AFIT/DS/ENG/98-04

# TRIGONOMETRIC TRANSFORMS FOR IMAGE RECONSTRUCTION

DISSERTATION

Thomas M. Foltz
Major, USAF

AFIT/DS/ENG/98-04

Approved for public release; distribution unlimited

19980629 032

DTIC QUALITY INSPECTED 1

The views expressed in this dissertation are those of the author and do not reflect the official policy or position of the Department of Defense or the United States Government.

AFIT/DS/ENG/98-04

# TRIGONOMETRIC TRANSFORMS FOR IMAGE RECONSTRUCTION

DISSERTATION

Presented to the Faculty of the Graduate School of Engineering

of the Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

Thomas M. Foltz, B.E.E., M.S.E.E.

Major, USAF

June 1998

Approved for public release; distribution unlimited

AFIT/DS/ENG/98-04

# TRIGONOMETRIC TRANSFORMS FOR IMAGE RECONSTRUCTION

Thomas M. Foltz, B.E.E., M.S.E.E.
Major, USAF

Approved: *Date:*

Byron M. Welsh — 21 May 98
Committee Chairman

Mark E. Oxley — 21 May 98
Committee Member

Michael C. Roggemann — 5/21/98
Committee Member

James T. Moore — 26 May 98
James T. Moore, LtCol, USAF
Dean's Representative

Accepted:

Robert A. Calico, Jr. — 26 May 98
Dean, Graduate School of Engineering

## *Acknowledgments*

I would first like to thank Chaplain, LtCol, Karen Chrisley, USAF, Ret., and the pastoral staff of Fairborn-St. Luke United Methodist Church for the spiritual guidance they provided during my time here in school.

I am indebted to the Air Force Maui Optical Station of the Air Force Materiel Command's Air Force Research Laboratory for sponsoring this research and to the Air Force Office of Scientific Research for additional funding they provided to our working group.

I owe a great deal to my advisor, Dr. Byron Welsh, for his superb mentorship, his caring guidance and his objective review of many documents surrounding this body of work. He allowed me a high degree of independence to pursue the course of this research for which I am grateful. I also extend thanks to the members of my committee, Dr. Mike Roggemann of Michigan Technological University for the many ideas he contributed, and Dr. Mark Oxley for challenging me to be thorough in my mathematical development.

Captains Steve Ford and Matt Whiteley deserve thanks for the many times I had to bother them for explanations and details of how things really worked. I also thank the many Comm Lab students who allowed me to share their workspace.

Finally and most importantly, I am grateful to my wife, Julie, first for a conversation we had over three years ago where she convinced me that this is what I really wanted to do, and also for the deeper appreciation I gained of my feelings for her during the course of this study. I cherish her kindness and commitment in addition to the support of our sons, Jim and Sam.

Tom Foltz

## *Table of Contents*

Page

## *List of Figures*

## *List of Tables*

## *Glossary of Notation*

| Symbol | Definition |
|---|---|
| $\circledast$ | convolution which is either circular or skew-circular |
| $\sim$ | extended version of a vector, matrix, or sequence |
| $\otimes$ | Kronecker or tensor product |
| $*$ | linear convolution |
| $\overline{\quad}$ | overbar denoting expectation |
| $\odot$ | point-wise or Hadamard product |
| $^*$ | superscript denoting complex conjugate |
| $\|\bullet\|_2$ | vector 2-norm |
| $\mathbf{0}$ | zero vector or zero matrix |
| $[\boldsymbol{A}]_{mn}$ | $m$-$n$th entry of the matrix $\boldsymbol{A}$ |
| $\textcircled{c}$ | circular convolution operator |
| $\boldsymbol{C}_{1e,N}$, ..., $\boldsymbol{C}_{4e,N}$ | $N \times N$ discrete cosine transform matrices of types I - IV for even-length sequences |
| $\boldsymbol{C}_{1o,N}$, ..., $\boldsymbol{C}_{4o,N}$ | $N \times N$ discrete cosine transform matrices of types I - IV for odd-length sequences |
| $\boldsymbol{C}_{\theta\theta}$ | object covariance matrix; $N \times N$ for one-dimensional object; $N_1N_2 \times N_1N_2$ for a lexicographically-ordered two-dimensional object |
| $\mathcal{C}_{1e}\{\bullet\}$, ..., $\mathcal{C}_{4e}\{\bullet\}$ | discrete cosine transform operators of types I - IV for even-length sequences |
| $\mathcal{C}_{1o}\{\bullet\}$, ..., $\mathcal{C}_{4o}\{\bullet\}$ | discrete cosine transform operators of types I - IV for odd-length sequences |

| Symbol | Definition |
|---|---|
| $d(n)$ | one-dimensional data sequence representing a corrupted, noisy version of a signal |
| $d(n_1,n_2)$ | two-dimensional data sequence representing a blurred, noisy image of an object |
| $D$ | arbitrary dimension; *e.g.*, a $D$-dimensional sequence, $x(n_1,n_2,\ldots,n_D)$ |
| $\boldsymbol{d}$ | data sequence expressed as a vector; $N \times 1$ for a one-dimensional sequence; $N_1 N_2 \times 1$ for a lexicographic representation of a two-dimensional sequence |
| $\boldsymbol{d}_a$, $\boldsymbol{d}_s$ | $N \times 1$ vectors which are portions of $\boldsymbol{d}$ calculated from symmetrically convolving the vectors $\boldsymbol{h}_a^r$ and $\boldsymbol{h}_s^r$ with the vector $\theta$; $\boldsymbol{d} = \boldsymbol{d}_a + \boldsymbol{d}_s$ |
| $\boldsymbol{d}'_a$, $\boldsymbol{d}'_s$ | $N \times 1$ vectors which are portions of $\boldsymbol{d}$ calculated from symmetrically convolving the vectors $\boldsymbol{h}_a^r$ and $\boldsymbol{h}_s^r$ with the vector $\theta$ using different forms of symmetric convolution; with appropriate zero-padding in the vectors $\boldsymbol{h}_a^r$, $\boldsymbol{h}_s^r$, and $\theta$, then $\boldsymbol{d}_a = \boldsymbol{d}'_a$ and $\boldsymbol{d}_s = \boldsymbol{d}'_s$ |
| $\boldsymbol{d}_{aa}$, $\boldsymbol{d}_{as}$, $\boldsymbol{d}_{sa}$, $\boldsymbol{d}_{ss}$ | $N_1 N_2 \times 1$ vectors which are portions of two-dimensional version of $\boldsymbol{d}$ calculated from symmetrically convolving the vectors $\boldsymbol{h}_{aa}^{rr}$, $\boldsymbol{h}_{as}^{rr}$, $\boldsymbol{h}_{sa}^{rr}$, and $\boldsymbol{h}_{ss}^{rr}$ with the vector $\theta$; $\boldsymbol{d} = \boldsymbol{d}_{aa} + \boldsymbol{d}_{as} + \boldsymbol{d}_{sa} + \boldsymbol{d}_{ss}$ |
| $\boldsymbol{d}'_{aa}$, $\boldsymbol{d}'_{as}$, $\boldsymbol{d}'_{sa}$, $\boldsymbol{d}'_{ss}$<br>$\boldsymbol{d}''_{aa}$, $\boldsymbol{d}''_{as}$, $\boldsymbol{d}''_{sa}$, $\boldsymbol{d}''_{ss}$<br>$\boldsymbol{d}'''_{aa}$, $\boldsymbol{d}'''_{as}$, $\boldsymbol{d}'''_{sa}$, $\boldsymbol{d}'''_{ss}$ | $N_1 N_2 \times 1$ vectors which are portions of two-dimensional version of $\boldsymbol{d}$ calculated from symmetrically convolving the vectors $\boldsymbol{h}_{aa}^{rr}$, $\boldsymbol{h}_{as}^{rr}$, $\boldsymbol{h}_{sa}^{rr}$, and $\boldsymbol{h}_{ss}^{rr}$ with the vector $\theta$ using different forms of symmetric convolution; with appropriate zero-padding in the vectors $\boldsymbol{h}_{aa}^{rr}$, $\boldsymbol{h}_{as}^{rr}$, $\boldsymbol{h}_{sa}^{rr}$, $\boldsymbol{h}_{ss}^{rr}$, and $\theta$, then $\boldsymbol{d}_{aa} = \boldsymbol{d}'_{aa} = \boldsymbol{d}''_{aa} = \boldsymbol{d}'''_{aa}$, $\boldsymbol{d}_{as} = \boldsymbol{d}'_{as} = \boldsymbol{d}''_{as} = \boldsymbol{d}'''_{as}$, $\boldsymbol{d}_{sa} = \boldsymbol{d}'_{sa} = \boldsymbol{d}''_{sa} = \boldsymbol{d}'''_{sa}$, and $\boldsymbol{d}_{ss} = \boldsymbol{d}'_{ss} = \boldsymbol{d}''_{ss} = \boldsymbol{d}'''_{ss}$ |
| $\boldsymbol{D}$ | two-dimensional data sequence expressed as an $N_1 \times N_2$ matrix |
| $\boldsymbol{D}_M$ | $M \times M$ diagonal matrix with elements $[\boldsymbol{D}]_{mm} = \exp\{j\frac{\pi}{M}m\}$ for $m = 0, 1, \ldots, M-1$; relates extended discrete trigonometric transform matrices to generalized discrete Fourier transform matrices |
| $E\{\bullet\}$ | expected value operator |

| Symbol | Definition |
|---|---|
| $\boldsymbol{E}_{SE}$ | shorthand notation to describe an arbitrary $M \times L$ symmetric extension matrix |
| $\boldsymbol{E}_{WSWS}$, ..., $\boldsymbol{E}_{HAWS}$ | $M \times L$ symmetric extension matrices; sixteen different matrices exist based on different possible ways to symmetrically extend a finite sequence |
| $(\boldsymbol{E}_c \otimes \boldsymbol{E}_r)$ | $M_1 M_2 \times L_1 L_2$ symmetric extension matrix which acts on a lexicographic vector representing a matrix; has the effect of extending both the rows and columns of the matrix; $\boldsymbol{E}_c$ and $\boldsymbol{E}_r$ are each one of the sixteen different symmetric extension matrices $\boldsymbol{E}_{WSWS}$, ..., $\boldsymbol{E}_{HAWS}$ |
| $\varepsilon_{SE}\{\bullet\}$ | shorthand notation to describe an arbitrary symmetric extension operator |
| $\varepsilon_{WSWS}\{\bullet\}$, ..., $\varepsilon_{HAWS}\{\bullet\}$ | symmetric extension operators; sixteen different operators exist based on different possible ways to symmetrically extend a finite sequence |
| $\varepsilon$ | error vector, $\varepsilon = \theta - \hat{\theta}$; $N \times 1$ if the vector $\theta$ represents a one-dimensional sequence; $N_1 N_2 \times 1$ if the vector $\theta$ represents a lexicographically-ordered two-dimensional sequence |
| $f(n)$, $f(n_1,n_2)$ | one and two-dimensional recovery filter impulse responses |
| $\boldsymbol{f}$ | vector representation of recovery filter impulse response; $N \times 1$ for a one-dimensional sequence; $N_1 N_2 \times 1$ for a lexicographically-ordered two-dimensional sequence |
| $\boldsymbol{F}$ | $N_1 \times N_2$ matrix representation of recovery filter two-dimensional impulse response |
| $\boldsymbol{F}_{BC}$ | $N_1 N_2 \times N_1 N_2$ block circulant matrix which performs two-dimensional circular convolution for recovery filter |
| $\boldsymbol{F}_{BSC}$ | $N_1 N_2 \times N_1 N_2$ block symmetric convolution matrix which performs two-dimensional symmetric convolution for recovery filter |

| Symbol | Definition |
|---|---|
| $\boldsymbol{F}_{BSC,aa}$, $\boldsymbol{F}_{BSC,as}$, $\boldsymbol{F}_{BSC,sa}$, $\boldsymbol{F}_{BSC,ss}$ | $N_1N_2 \times N_1N_2$ block symmetric convolution matrices implementing two-dimensional block symmetric convolution by the antisymmetric and symmetric portions of the matrix $\boldsymbol{F}_{BSC}$; the *'BSC'* subscript is dropped in later chapters when it is obvious that the form of convolution is block symmetric |
| $\boldsymbol{F}_C$ | $N \times N$ circulant matrix which performs one-dimensional circular convolution for recovery filter |
| $\boldsymbol{F}_{SC}$ | $N \times N$ symmetric convolution matrix which performs one-dimensional symmetric convolution for recovery filter |
| $\boldsymbol{F}_{SC,a}$, $\boldsymbol{F}_{SC,s}$ | $N \times N$ symmetric convolution matrices implementing symmetric convolution by the antisymmetric and symmetric portions of the matrix $\boldsymbol{F}_{SC}$; the *'SC'* subscript is dropped in later chapters when it is obvious that the form of convolution is symmetric |
| $\mathcal{F}\{\bullet\}$ | discrete Fourier transform operator |
| $\mathcal{F}_F(k)$, $\mathcal{F}_F(k_1,k_2)$ | discrete Fourier transforms of $f(n)$ and $f(n_1,n_2)$ |
| $\mathcal{F}_T(k)$, $\mathcal{F}_T(k_1,k_2)$ | discrete trigonometric transforms of $f(n)$ and $f(n_1,n_2)$ |
| $\mathcal{F}_{T,a}(k)$, $\mathcal{F}_{T,s}(k)$ | decomposition of $\mathcal{F}_T(k)$ into its antisymmetric and symmetric parts; the *'T'* subscript is dropped in later chapters when it is obvious that the transform domain is trigonometric |
| $\mathcal{F}_{T,aa}(k_1,k_2)$, $\mathcal{F}_{T,as}(k_1,k_2)$, $\mathcal{F}_{T,sa}(k_1,k_2)$, $\mathcal{F}_{T,ss}(k_1,k_2)$ | decomposition of $\mathcal{F}_T(k_1,k_2)$ into its antisymmetric and symmetric parts; the *'T'* subscript is dropped in later chapters when it is obvious that the transform domain is trigonometric |
| $\boldsymbol{\mathcal{F}}_F$ | diagonalized recovery filter matrix in the Fourier domain; $N \times N$ for one-dimensional data; $N_1N_2 \times N_1N_2$ for lexicographically-ordered two-dimensional data |
| $\boldsymbol{\mathcal{F}}_T$ | diagonalized recovery filter matrix in the trigonometric transform domain; $N \times N$ for one-dimensional data; $N_1N_2 \times N_1N_2$ for lexicographically-ordered two-dimensional data |

| Symbol | Definition |
|---|---|
| $\mathcal{F}_{T,a}$, $\mathcal{F}_{T,s}$ | $N \times N$ diagonalized forms of one-dimensional symmetric convolution matrices $\boldsymbol{F}_{SC,a}$ and $\boldsymbol{F}_{SC,s}$; the *'T'* subscript is dropped in later chapters when it is obvious that the transform domain is trigonometric |
| $\mathcal{F}_{T,aa}$, $\mathcal{F}_{T,as}$, $\mathcal{F}_{T,sa}$, $\mathcal{F}_{T,ss}$ | $N_1N_2 \times N_1N_2$ trigonometric transforms of the matrices $\boldsymbol{F}_{BSC,aa}$, $\boldsymbol{F}_{BSC,as}$, $\boldsymbol{F}_{BSC,sa}$, and $\boldsymbol{F}_{BSC,ss}$; the *'T'* subscript is dropped in later chapters when it is obvious that the transform domain is trigonometric |
| $\boldsymbol{G}_{a,b,M}$ | $M \times M$ generalized discrete Fourier transform matrix with input and output index variable shifts of $a$ and $b$ |
| $\mathcal{G}_{a,b}\{\bullet\}$ | generalized discrete Fourier transform operator with input and output index variable shifts of $a$ and $b$ |
| $h(n)$ | one-dimensional degrading system impulse response |
| $h(n_1,n_2)$ | two-dimensional degrading system point spread function |
| $h_a(n)$, $h_s(n)$ | antisymmetric and symmetric portions of $h(n)$; $h(n) = h_a(n) + h_s(n)$ |
| $h_a^r(n)$, $h_s^r(n)$ | sequences representing portions of $h_a(n)$ and $h_s(n)$ which exist to the right of the origin; the superscript *'r'* indicates the filter right-half terms |
| $h_{aa}(n_1,n_2)$, $h_{as}(n_1,n_2)$, $h_{sa}(n_1,n_2)$, $h_{ss}(n_1,n_2)$ | sequences representing portions of the decomposition of $h(n_1,n_2)$ into its antisymmetric and symmetric parts about the $n_1$ and $n_2$ axes; $h(n_1,n_2) = h_{aa}(n_1,n_2) + h_{as}(n_1,n_2) + h_{sa}(n_1,n_2) + h_{ss}(n_1,n_2)$ |
| $h_{aa}^{rr}(n_1,n_2)$, $h_{as}^{rr}(n_1,n_2)$, $h_{sa}^{rr}(n_1,n_2)$, $h_{ss}^{rr}(n_1,n_2)$ | sequences representing portions of $h_{aa}(n_1,n_2)$, $h_{as}(n_1,n_2)$, $h_{sa}(n_1,n_2)$, and $h_{ss}(n_1,n_2)$ which exist in the first quadrant of the $n_1$ - $n_2$ plane; the superscript *'rr'* indicates the filter right-half terms about both $n_1$ and $n_2$ |
| $\boldsymbol{h}$ | vector representation of degrading system impulse response; $N \times 1$ for a one-dimensional sequence; $N_1N_2 \times 1$ for a lexicographically-ordered two-dimensional sequence |

| Symbol | Definition |
|---|---|
| $\boldsymbol{h}_a^r$, $\boldsymbol{h}_s^r$ | $N \times 1$ vector representations of $h_a^r(n)$ and $h_s^r(n)$ |
| $\boldsymbol{h}_{aa}^{rr}$, $\boldsymbol{h}_{as}^{rr}$, $\boldsymbol{h}_{sa}^{rr}$, $\boldsymbol{h}_{ss}^{rr}$ | $N_1N_2 \times 1$ lexicographic vector representations of $h_{aa}^{rr}(n_1,n_2)$, $h_{as}^{rr}(n_1,n_2)$, $h_{sa}^{rr}(n_1,n_2)$, and $h_{ss}^{rr}(n_1,n_2)$ |
| $H$ | superscript denoting Hermitian or conjugate transpose |
| $\boldsymbol{H}$ | $N_1 \times N_2$ matrix representation of two-dimensional degrading system point spread function |
| $\boldsymbol{H}_{aa}^{rr}$, $\boldsymbol{H}_{as}^{rr}$, $\boldsymbol{H}_{sa}^{rr}$, $\boldsymbol{H}_{ss}^{rr}$ | $N_1 \times N_2$ matrix representations of $h_{aa}^{rr}(n_1,n_2)$, $h_{as}^{rr}(n_1,n_2)$, $h_{sa}^{rr}(n_1,n_2)$, and $h_{ss}^{rr}(n_1,n_2)$ |
| $\boldsymbol{H}_B$ | $M_1M_2 \times M_1M_2$ matrix which can be either block circulant, block skew-circulant, or a combination depending on the underlying symmetry of the rows and columns of $\boldsymbol{H}$ |
| $\boldsymbol{H}_{BC}$ | $N_1N_2 \times N_1N_2$ block circulant matrix which performs two-dimensional circular convolution for degrading system |
| $\boldsymbol{H}_{BS}$ | $N_1N_2 \times N_1N_2$ block skew-circulant matrix which performs two-dimensional skew-circular convolution |
| $\boldsymbol{H}_{BSC}$ | $N_1N_2 \times N_1N_2$ block symmetric convolution matrix |
| $\boldsymbol{H}_{BSC,aa}$, $\boldsymbol{H}_{BSC,as}$, $\boldsymbol{H}_{BSC,sa}$, $\boldsymbol{H}_{BSC,ss}$ | $N_1N_2 \times N_1N_2$ block symmetric convolution matrices implementing two-dimensional block symmetric convolution by the matrices $\boldsymbol{H}_{aa}^{rr}$, $\boldsymbol{H}_{as}^{rr}$, $\boldsymbol{H}_{sa}^{rr}$, and $\boldsymbol{H}_{ss}^{rr}$; the *'BSC'* subscript is dropped in later chapters when it is obvious that the form of convolution is block symmetric |
| $\boldsymbol{H}'_{BSC,aa}$, ..., $\boldsymbol{H}'_{BSC,ss}$ $\boldsymbol{H}''_{BSC,aa}$, ..., $\boldsymbol{H}''_{BSC,ss}$ $\boldsymbol{H}'''_{BSC,aa}$, ..., $\boldsymbol{H}'''_{BSC,ss}$ | $N_1N_2 \times N_1N_2$ block symmetric convolution matrices implementing two-dimensional block symmetric convolution by the matrices $\boldsymbol{H}_{aa}^{rr}$, $\boldsymbol{H}_{as}^{rr}$, $\boldsymbol{H}_{sa}^{rr}$, and $\boldsymbol{H}_{ss}^{rr}$ using alternate but equivalent forms of block symmetric convolution |
| $\boldsymbol{H}_C$ | $N \times N$ circulant matrix which performs one-dimensional circular convolution for degrading system |

| Symbol | Definition |
|---|---|
| $\boldsymbol{H}_{C\times S}$ | $N_1N_2 \times N_1N_2$ matrix of partitioned blocks of skew-circulant matrices arranged in a circulant pattern |
| $\boldsymbol{H}_M$ | $M \times M$ circulant or skew-circulant matrix used inside matrix definition for symmetric convolution |
| $\boldsymbol{H}_S$ | $N \times N$ skew-circulant matrix which performs one-dimensional skew-circular convolution |
| $\boldsymbol{H}_{S\times C}$ | $N_1N_2 \times N_1N_2$ matrix of partitioned blocks of circulant matrices arranged in a skew-circulant pattern |
| $\boldsymbol{H}_{SC}$ | $N \times N$ symmetric convolution matrix |
| $\boldsymbol{H}_{SC,a}$, $\boldsymbol{H}_{SC,s}$ | $N \times N$ symmetric convolution matrices implementing symmetric convolution by the vectors $\boldsymbol{h}_a^r$ and $\boldsymbol{h}_s^r$; *e.g.*, $\boldsymbol{H}_{SC,a} = \boldsymbol{S}_{2e,N}^{-1}\boldsymbol{\mathcal{H}}_{T,a}\boldsymbol{C}_{2e,N}$ and $\boldsymbol{H}_{SC,s} = \boldsymbol{C}_{2e,N}^{-1}\boldsymbol{\mathcal{H}}_{T,s}\boldsymbol{C}_{2e,N}$; the *'SC'* subscript is dropped in later chapters when it is obvious that the form of convolution is symmetric |
| $\boldsymbol{H}'_{SC,a}$, $\boldsymbol{H}'_{SC,s}$ | $N \times N$ symmetric convolution matrices implementing symmetric convolution by the vectors $\boldsymbol{h}_a^r$ and $\boldsymbol{h}_s^r$ using alternate but equivalent forms of symmetric convolution; *e.g.*, $\boldsymbol{H}'_{SC,a} = -\boldsymbol{C}_{2e,N}^{-1}\boldsymbol{\mathcal{H}}_{T,a}\boldsymbol{S}_{2e,N}$ and $\boldsymbol{H}'_{SC,s} = \boldsymbol{S}_{2e,N}^{-1}\boldsymbol{\mathcal{H}}_{T,s}\boldsymbol{S}_{2e,N}$; $\boldsymbol{H}_{SC,a} \neq \boldsymbol{H}'_{SC,a}$ and $\boldsymbol{H}_{SC,s} \neq \boldsymbol{H}'_{SC,s}$ |
| $\mathcal{H}_F(k)$, $\mathcal{H}_F(k_1,k_2)$ | one and two-dimensional discrete Fourier transforms of $h(n)$ and $h(n_1,n_2)$ |
| $\mathcal{H}_O(k)$, $\mathcal{H}_O(k_1,k_2)$ | one and two-dimensional odd discrete Fourier transforms of $h(n)$ and $h(n_1,n_2)$ |
| $\mathcal{H}_T(k)$, $\mathcal{H}_T(k_1,k_2)$ | one and two-dimensional discrete trigonometric transforms of $h(n)$ and $h(n_1,n_2)$ |
| $\mathcal{H}_{T,a}(k)$, $\mathcal{H}_{T,s}(k)$ | decomposition of $\mathcal{H}_T(k)$ into its antisymmetric and symmetric parts; the *'T'* subscript is dropped in later chapters when it is obvious that the transform domain is trigonometric |

| Symbol | Definition |
|---|---|
| $\mathcal{H}_{T,aa}(k_1,k_2)$, $\mathcal{H}_{T,as}(k_1,k_2)$, $\mathcal{H}_{T,sa}(k_1,k_2)$, $\mathcal{H}_{T,ss}(k_1,k_2)$ | decomposition of $\mathcal{H}_T(k_1,k_2)$ into its antisymmetric and symmetric parts; the '$T$' subscript is dropped in later chapters when it is obvious that the transform domain is trigonometric |
| $\boldsymbol{h}_{aa}^{rr}$, $\boldsymbol{h}_{as}^{rr}$, $\boldsymbol{h}_{sa}^{rr}$, $\boldsymbol{h}_{ss}^{rr}$ | $N_1N_2 \times 1$ trigonometric transforms of the vectors $\boldsymbol{h}_{aa}^{rr}$, $\boldsymbol{h}_{as}^{rr}$, $\boldsymbol{h}_{sa}^{rr}$, and $\boldsymbol{h}_{ss}^{rr}$ |
| $\mathcal{H}_F$ | diagonalized degrading system convolution matrix in the Fourier domain; $N \times N$ for a one-dimensional object; $N_1N_2 \times N_1N_2$ for a lexicographically-ordered two-dimensional object |
| $\mathcal{H}_{F\times O}$ | $N_1N_2 \times N_1N_2$ diagonal form of $\boldsymbol{H}_{C\times S}$ |
| $\mathcal{H}_M$ | $M \times M$ transform of $\boldsymbol{H}_M$; equal to $\mathcal{H}_F$ if $\boldsymbol{H}_M$ is circulant, or $\mathcal{H}_O$ if $\boldsymbol{H}_M$ is skew-circulant |
| $\mathcal{H}_O$ | diagonalized convolution matrix in the odd Fourier transform domain; $N \times N$ for a one-dimensional object; $N_1N_2 \times N_1N_2$ for a lexicographically-ordered two-dimensional object |
| $\mathcal{H}_{O\times F}$ | $N_1N_2 \times N_1N_2$ diagonal form of $\boldsymbol{H}_{S\times C}$ |
| $\mathcal{H}_T$ | diagonalized symmetric convolution matrix in the trigonometric transform domain; $N \times N$ for a one-dimensional object; $N_1N_2 \times N_1N_2$ for a lexicographically-ordered two-dimensional object |
| $\mathcal{H}_{T,a}$, $\mathcal{H}_{T,s}$ | $N \times N$ diagonalized forms of one-dimensional symmetric convolution matrices; *e.g.*, $\mathcal{H}_{T,a} = \text{diag}\{\boldsymbol{S}_{1e,N}\boldsymbol{h}_a^r\}$ and $\mathcal{H}_{T,s} = \text{diag}\{\boldsymbol{C}_{1e,N}\boldsymbol{h}_s^r\}$; the '$T$' subscript is dropped in later chapters when it is obvious that the transform domain is trigonometric |
| $\mathcal{H}_{T,aa}^{rr}$, $\mathcal{H}_{T,as}^{rr}$, $\mathcal{H}_{T,sa}^{rr}$, $\mathcal{H}_{T,ss}^{rr}$ | $N_1 \times N_2$ trigonometric transforms of the matrices $\boldsymbol{H}_{aa}^{rr}$, $\boldsymbol{H}_{as}^{rr}$, $\boldsymbol{H}_{sa}^{rr}$, and $\boldsymbol{H}_{ss}^{rr}$ |
| $\mathcal{H}_{T,aa}$, $\mathcal{H}_{T,as}$, $\mathcal{H}_{T,sa}$, $\mathcal{H}_{T,ss}$ | $N_1N_2 \times N_1N_2$ trigonometric transforms of the matrices $\boldsymbol{H}_{BSC,aa}$, $\boldsymbol{H}_{BSC,as}$, $\boldsymbol{H}_{BSC,sa}$, and $\boldsymbol{H}_{BSC,ss}$; the '$T$' subscript is dropped in later chapters when it is obvious that the transform domain is trigonometric |

| Symbol | Definition |
|---|---|
| $\boldsymbol{I}$ | identity matrix when dimension is obvious from context |
| $\boldsymbol{I}_L$ | $L \times L$ identity matrix |
| $k$ | one-dimensional transform domain index |
| $k_1$, $k_2$ | two-dimensional transform domain indices |
| $k_n$ | constant term in definition of discrete trigonometric transform matrices; $k_n = 1/2$ for $n = 0$ or $N$, 1 otherwise |
| $L$ | length of possibly $N-1$, $N$, or $N+1$ depending on length of particular discrete trigonometric transform |
| $L_1$, $L_2$ | lengths of possibly $N_1-1$, $N_1$, or $N_1+1$ and $N_2-1$, $N_2$, or $N_2+1$ depending on length of particular two-dimensional discrete trigonometric transform |
| $\min\{\bullet\}$ | minimum |
| $M$ | length of a extended version of a one-dimensional sequence; $M = 2N$ for $N$ even or $M = 2N-1$ for $N$ odd |
| $M_1$, $M_2$ | lengths of extended versions of two-dimensional sequences; $M_1 = 2N_1$ for $N_1$ even or $M_1 = 2N_1 - 1$ for $N_1$ odd, and $M_2 = 2N_2$ for $N_2$ even or $M_2 = 2N_2 - 1$ for $N_2$ odd |
| $\mu_\theta$ | object mean vector; $N \times 1$ for a one-dimensional object; $N_1 N_2 \times 1$ for a lexicographically-ordered two-dimensional object |
| $n$ | one-dimensional sequence domain index |
| $n_0$ | a particular value of $n$; symmetric convolution induces a delay of $n_0 = 0$ or 1 |
| $n_1$, $n_2$ | two-dimensional spatial domain indices |
| $N$ | length of a one-dimensional sequence |
| $N_1$, $N_2$ | lengths in $n_1$, $n_2$ directions of two-dimensional sequences |

| Symbol | Definition |
|---|---|
| $\mathcal{R}_L(n)$ | rectangular window of length $L$ |
| $\boldsymbol{R}_L$ | $L \times M$ matrix equivalent of $\mathcal{R}_L(n)$; $\boldsymbol{R}_L = \left[\boldsymbol{I}_L \;\; \boldsymbol{0}\right]$ |
| $\left(\boldsymbol{R}_{L_1} \otimes \boldsymbol{R}_{L_2}\right)$ | $L_1 L_2 \times M_1 M_2$ windowing matrix which retains samples of interest form two-dimensional symmetric convolution expression; $\boldsymbol{R}_{L_1}$ retains $L_1$ samples from each column, and $\boldsymbol{R}_{L_2}$ retains $L_2$ samples from each row; $\boldsymbol{R}_{L_1} = \left[\boldsymbol{I}_{L_1} \;\; \boldsymbol{0}\right]$ and $\boldsymbol{R}_{L_2} = \left[\boldsymbol{I}_{L_2} \;\; \boldsymbol{0}\right]$ |
| $\boldsymbol{R}_{\theta\theta}$ | object correlation matrix; $N \times N$ for a one-dimensional object; $N_1 N_2 \times N_1 N_2$ for a lexicographically-ordered two-dimensional object |
| $\boldsymbol{R}_{\Theta_F \Theta_F}$ | Fourier domain object correlation matrix; $N \times N$ for a one-dimensional object; $N_1 N_2 \times N_1 N_2$ for a lexicographically-ordered two-dimensional object |
| $\boldsymbol{R}_{\Theta_s \Theta_s}$ | $N \times N$ discrete cosine transform domain object correlation matrix for a one-dimensional object; subscript *'s'* denotes symmetric portion |
| $\boldsymbol{R}_{\Theta_{ss} \Theta_{ss}}$ | $N_1 N_2 \times N_1 N_2$ discrete cosine transform domain object correlation matrix for a lexicographically-ordered two-dimensional object; subscript *'ss'* denotes symmetric portion about both axes |
| $\boldsymbol{R}_{ww}$ | noise correlation matrix; $N \times N$ for a one-dimensional noise sequence; $N_1 N_2 \times N_1 N_2$ for a lexicographically-ordered two-dimensional noise sequence |
| $\boldsymbol{R}_{w_a w_a}$, $\boldsymbol{R}_{w_s w_s}$ | $N \times N$ trigonometric domain noise correlation matrices for a one-dimensional noise sequence; subscripts *'a'* and *'s'* denote antisymmetric and symmetric portions |
| $\boldsymbol{R}_{w_{aa} w_{aa}}$, $\boldsymbol{R}_{w_{as} w_{as}}$, $\boldsymbol{R}_{w_{sa} w_{sa}}$, $\boldsymbol{R}_{w_{ss} w_{ss}}$ | $N_1 N_2 \times N_1 N_2$ trigonometric domain noise correlation matrices for a lexicographically-ordered two-dimensional noise sequence; subscripts *'aa'*, *'as'*, *'sa'*, and *'ss'* denote antisymmetric and symmetric portions about the $n_1$ and $n_2$ axes |

| Symbol | Definition |
|---|---|
| $\boldsymbol{R}_{w_F w_F}$ | Fourier domain noise correlation matrix; $N \times N$ for one-dimensional data; $N_1 N_2 \times N_1 N_2$ for lexicographically-ordered two-dimensional data |
| $\rho$ | correlation coefficient |
| $\circledS$ | skew-circular convolution operator |
| $\boldsymbol{S}_{1e,N}$, ..., $\boldsymbol{S}_{4e,N}$ | $N \times N$ discrete sine transform matrices of types I - IV for even-length sequences |
| $\boldsymbol{S}_{1o,N}$, ..., $\boldsymbol{S}_{4o,N}$ | $N \times N$ discrete sine transform matrices of types I - IV for odd-length sequences |
| $\mathcal{S}_{1e}\{\bullet\}$, ..., $\mathcal{S}_{4e}\{\bullet\}$ | discrete sine transform operators of types I - IV for even-length sequences |
| $\mathcal{S}_{1o}\{\bullet\}$, ..., $\mathcal{S}_{4o}\{\bullet\}$ | discrete sine transform operators of types I - IV for odd-length sequences |
| $^T$ | superscript denoting transpose of a vector or matrix |
| $\mathrm{Tr}\{\bullet\}$ | trace of a matrix |
| $\boldsymbol{T}_L$ | $L \times L$ trigonometric transform matrix |
| $\tilde{\boldsymbol{T}}$ | $M \times L$ extended trigonometric transform matrix; $\tilde{\boldsymbol{T}}$ has twice as many rows as $\boldsymbol{T}_L$ |
| $\left(\boldsymbol{T}_{c,L_1} \otimes \boldsymbol{T}_{r,L_2}\right)$ | $L_1 L_2 \times L_1 L_2$ two-dimensional trigonometric transform matrix which acts on a lexicographic vector representing a matrix; $\boldsymbol{T}_{c,L_1}$ and $\boldsymbol{T}_{r,L_2}$ are different one-dimensional trigonometric transform matrices which act on the columns and the rows of the matrix |
| $\mathcal{T}\{\bullet\}$ | trigonometric transform operator |
| $\theta(n)$, $\theta(n_1,n_2)$ | one and two-dimensional object sequences |
| $\hat{\theta}(n)$, $\hat{\theta}(n_1,n_2)$ | one and two-dimensional object estimates |

| Symbol | Definition |
|---|---|
| $\mathcal{J}_F(k),\ \mathcal{J}_F(k_1,k_2)$ | one and two-dimensional discrete Fourier transforms of an object sequence |
| $\mathcal{J}_O(k),\ \mathcal{J}_O(k_1,k_2)$ | one and two-dimensional odd discrete Fourier transforms of an object sequence |
| $\mathcal{J}_T(k),\ \mathcal{J}_T(k_1,k_2)$ | one and two-dimensional discrete trigonometric transforms of an object sequence |
| $\overline{\mathcal{O}_F^2(k)},\ \overline{\mathcal{O}_F^2(k_1,k_2)}$ | diagonal elements of one and two-dimensional Fourier domain object correlation matrices |
| $\overline{\mathcal{O}_s^2(k)},\ \overline{\mathcal{O}_{ss}^2(k_1,k_2)}$ | diagonal elements of one and two-dimensional discrete cosine transform domain object correlation matrices |
| $\theta$ | object sequence expressed as a vector; $N \times 1$ for a one-dimensional object; $N_1 N_2 \times 1$ for a lexicographically-ordered two-dimensional object |
| $\hat{\theta}$ | object estimate expressed as a vector; $N \times 1$ for a one-dimensional object; $N_1 N_2 \times 1$ for a lexicographically-ordered two-dimensional object |
| $\hat{\theta}_{\min}$ | linear minimum mean-squared error estimate; $N \times 1$ for a one-dimensional object; $N_1 N_2 \times 1$ for a lexicographically-ordered two-dimensional object |
| $\mathcal{J}_F$ | discrete Fourier transform of object vector; $N \times 1$ for a one-dimensional object; $N_1 N_2 \times 1$ for a lexicographically-ordered two-dimensional object |
| $\mathcal{J}_T$ | discrete trigonometric transform of object vector; $N \times 1$ for a one-dimensional object; $N_1 N_2 \times 1$ for a lexicographically-ordered two-dimensional object |
| $\mathcal{J}_{T,ss}$ | $N_1 N_2 \times 1$ vector in the discrete cosine transform domain having half-sample symmetry about both axes |
| $\mathcal{O}$ | $N_1 \times N_2$ matrix representation of a two-dimensional object |

| Symbol | Definition |
|---|---|
| $\mathcal{O}_F$ | $N_1 \times N_2$ matrix representation of discrete Fourier transform of object matrix |
| $\mathcal{O}_T$ | $N_1 \times N_2$ matrix representation of discrete trigonometric transform of object matrix |
| $\mathcal{O}_{T,ss}$ | $N_1 \times N_2$ matrix representation of $\vartheta_{T,ss}$ |
| $\text{vec}\{\bullet\}$ | converts a matrix into a lexicographically-ordered vector |
| $\boldsymbol{V}_N^{-1}$ | $N \times N$ odd discrete Fourier transform matrix; $\boldsymbol{V}_N^{-1} = \boldsymbol{G}_{\frac{1}{2},0,N}$ |
| $w(n),\ w(n_1,n_2)$ | one and two-dimensional noise sequences |
| $\overline{\mathcal{W}_F^2(k)},\ \overline{\mathcal{W}_F^2(k_1,k_2)}$ | diagonal elements of one and two-dimensional Fourier domain noise correlation matrices |
| $\overline{\mathcal{W}_a^2(k)},\ \overline{\mathcal{W}_s^2(k)}$ | diagonal elements of one-dimensional trigonometric transform domain noise correlation matrices, $\boldsymbol{R}_{\mathcal{W}_a\mathcal{W}_a}$, and $\boldsymbol{R}_{\mathcal{W}_s\mathcal{W}_s}$ |
| $\overline{\mathcal{W}_{aa}^2(k_1,k_2)},\ \overline{\mathcal{W}_{as}^2(k_1,k_2)},$ $\overline{\mathcal{W}_{sa}^2(k_1,k_2)},\ \overline{\mathcal{W}_{ss}^2(k_1,k_2)}$ | diagonal elements of one-dimensional trigonometric transform domain noise correlation matrices, $\boldsymbol{R}_{\mathcal{W}_{aa}\mathcal{W}_{aa}}$, $\boldsymbol{R}_{\mathcal{W}_{as}\mathcal{W}_{as}}$, $\boldsymbol{R}_{\mathcal{W}_{sa}\mathcal{W}_{sa}}$, and $\boldsymbol{R}_{\mathcal{W}_{ss}\mathcal{W}_{ss}}$ |
| $\boldsymbol{W}_N^{-1}$ | $N \times N$ discrete Fourier transform matrix |
| $x(n),\ x(n_1,n_2)$ | arbitrary one and two-dimensional sequences |
| $x_a(n),\ x_s(n)$ | antisymmetric and symmetric portions of $x(n)$; $x(n) = x_a(n) + x_s(n)$ |
| $x_{aa}(n_1,n_2),\ x_{as}(n_1,n_2),$ $x_{sa}(n_1,n_2),\ x_{ss}(n_1,n_2)$ | sequences representing portions of the decomposition of $x(n_1,n_2)$ into its antisymmetric and symmetric parts about the $n_1$ and $n_2$ axes; $x(n_1,n_2) = x_{aa}(n_1,n_2) + x_{as}(n_1,n_2) + x_{sa}(n_1,n_2) + x_{ss}(n_1,n_2)$ |
| $\boldsymbol{x}$ | arbitrary one-dimensional sequence expressed as an $N \times 1$ vector |
| $\tilde{\boldsymbol{x}}$ | $M \times 1$ symmetric extension of $\boldsymbol{x}$ |

## *Abstract*

This dissertation demonstrates how the symmetric convolution-multiplication property of discrete trigonometric transforms can be applied to traditional problems in image reconstruction with slightly better performance than Fourier techniques and increased savings in computational complexity for symmetric point spread functions. The fact that the discrete Fourier transform diagonalizes a circulant matrix provides an alternate way to derive the symmetric convolution-multiplication property for discrete trigonometric transforms. Derived in this manner, the symmetric convolution-multiplication property extends easily to multiple dimensions and generalizes to multidimensional asymmetric sequences. The symmetric convolution-multiplication property allows for linear filtering of degraded images via point-by-point multiplication in the transform domain of trigonometric transforms. Specifically in the transform domain of a type-II discrete cosine transform, there is an asymptotically optimum energy compaction about the low-frequency indices of highly correlated images which has advantages in reconstructing images with high-frequency noise. The symmetric convolution-multiplication property allows for well-approximated scalar representations in the trigonometric transform domain for linear reconstruction filters such as the Wiener filter. An analysis of the scalar Wiener filter's improved mean-squared error performance in the trigonometric transform domain is given.

# TRIGONOMETRIC TRANSFORMS FOR IMAGE RECONSTRUCTION

## I. *Introduction*

The results of the research presented in this dissertation demonstrate new forms of linear image reconstruction filters which employ a recently-developed property of trigonometric transforms. Image reconstruction is the process of restoring degraded images [40]. An imaging system typically measures a blurred, noisy image of an object. The system must then apply image reconstruction techniques to recover an estimate of the object from its blurred, noisy version. Linear image reconstruction techniques use linear shift-invariant filters to recover the object [27]. A linear shift-invariant filter is a system whose response to an arbitrary input is completely characterized by its response to a single impulse [37].

The Air Force is interested in image reconstruction because it has a need to image spaceborne objects from the ground [13]. This problem is complicated by the turbulent atmosphere which has a severe degrading effect on the quality of images [40] and by the noise present in image detection systems [17]. Many existing nonlinear image reconstruction techniques are iterative [30] - [32], and require large amounts of computer processing time with a high degree of computational complexity. Existing linear methods [15], [21], [27], [38], [44], require fewer computations than iterative methods, but suffer from poorer performance. An image reconstruction technique which has better performance than existing linear techniques, yet still offers the computational advantage of linearity would thus be of great interest to the Air Force.

The general purpose of this dissertation is to investigate the application of trigonometric transforms to the two-dimensional image reconstruction problem. Two-dimensional trigonometric transforms [26], [27] convert the pixels of a discretely-sampled image into a two-dimensional series of coefficients which represents the amount of information contained in the image at sinusoids of different spatial frequencies. A spatial frequency is the two-dimensional equivalent of a one-dimensional temporal frequency. Specifically, this research shows how the symmetric convolution-multiplication property of discrete trigonometric transforms can improve linear image reconstruction filters.

Martucci [34] recently developed the symmetric convolution-multiplication property for the family of discrete trigonometric transforms. The discrete trigonometric transform family consists of sixteen different one-dimensional transforms which are even and odd-length versions of type I - IV discrete sine and cosine transforms [26], [27], [39]. A sine or cosine transform performs a similar operation on an image as a Fourier transform. Consider the sequence $\vartheta_T(k_1,k_2) = \mathcal{T}\{\theta(n_1,n_2)\}$ which is the two-dimensional trigonometric transform of the sequence $\theta(n_1,n_2)$. The subscript '$T$' denotes that the sequence $\vartheta_T(k_1,k_2)$ exists in the trigonometric transform domain. The operator $\mathcal{T}\{\bullet\}$ is a two-dimensional discrete trigonometric transform based on any of the sixteen one-dimensional discrete trigonometric transforms. The indices $n_1$ and $n_2$ correspond to an ordering of the pixels of the image. The indices $k_1$ and $k_2$ represent the locations of sample points in trigonometric transform domain space which correspond to different spatial frequencies. The values of the sequence $\vartheta_T(k_1,k_2)$ at each point represent the amount of information contained in the image which agrees with sinusoids having spatial frequencies corresponding to $k_1$ and $k_2$.

Similarly the sequence $\vartheta_F(k_1,k_2) = \mathcal{F}\{\theta(n_1,n_2)\}$ in the Fourier domain represents the amount of information contained in an image which agrees with complex exponentials of different spatial frequencies. Here the subscript *'F'* indicates the Fourier domain, and the operator $\mathcal{F}\{\bullet\}$ represents the discrete Fourier transform. The sequences $\vartheta_T(k_1,k_2)$ and $\vartheta_F(k_1,k_2)$ are both transform-domain spatial frequency representations of the sequence $\theta(n_1,n_2)$. One notable difference between the sequences $\vartheta_T(k_1,k_2)$ and $\vartheta_F(k_1,k_2)$ is that if $\theta(n_1,n_2)$ is real-valued, then the sequence $\vartheta_T(k_1,k_2)$ will also be real-valued, but the sequence $\vartheta_F(k_1,k_2)$ will be complex-valued.

There are sixteen different one-dimensional discrete trigonometric transforms. Different transforms exist for sines and cosines, for even and odd-length sequences, and for different types distinguished as types I - IV [26], [27], [39]. The four types I - IV impose half-sample shifts in the input or output indices of the sine and cosine transforms. A type-I trigonometric transform imposes no shift to either the input or the output sequence indices. A type-II transform imposes a half-sample shift to just the input index. A type-III transform imposes a half-sample shift to just the output index. A type-IV transform imposes a half-sample shift to both the input and the output indices. The fact that these four transforms require either no shift or a half-sample shift is related to the idea of the point of symmetry in the symmetric extension of a finite sequence. The point of symmetry can be either the end point in the sequence or a point which lies one-half sample beyond the end point of the sequence.

The discrete trigonometric transform family lies at the heart of many image transform coding applications [27], [33], [39]. The objective of image transform coding, which is different than the goal of this research, is to reduce the information content of an image for storage or transmission. Although very useful for transform coding, the discrete trigonometric transforms

have not proved very useful in a wide variety of image filtering applications, because until recently no convolution-multiplication property existed for the entire family. The advantage of a transform that possesses a convolution-multiplication property is that the property allows a filter to be implemented more efficiently in the transform domain through point-wise multiplication. To implement the effects of a filter through convolution directly in the spatial domain requires a large number of shifts, adds, and multiplies.

The circular convolution-multiplication property of discrete Fourier transforms is well-known [37]. A symmetric convolution-multiplication property for the entire family of discrete trigonometric transforms has only existed since 1994 [34]. An earlier attempt to define a convolution-multiplication property for the discrete cosine transform [6] produced a result limited to the convolution of only certain types of symmetric sequences. It was based solely on the type-II even-length discrete cosine transform.

Martucci [34] defines symmetric convolution as the form of convolution for the entire family of discrete trigonometric transforms similar to the manner in which circular convolution [37] is the form of convolution for the discrete Fourier transform. Circular convolution involves the point-by-point shifting and adding of periodic replicas of the finite sequences being convolved. Symmetric convolution involves the point-by-point shifting and adding of symmetric extensions of the sequences being convolved. The symmetric convolution-multiplication property states that an inverse trigonometric transform of the product of the trigonometric transforms of two sequences yields the same result as the symmetric convolution of the two sequences [34]. Mathematically, the symmetric convolution-multiplication property states that the sequence $d(n_1,n_2)$ which is calculated as $d(n_1,n_2) = \mathcal{T}_3^{-1}\{\mathcal{T}_1\{\theta(n_1,n_2)\}\cdot\mathcal{T}_2\{h(n_1,n_2)\}\}$ is exactly the same sequence which results from computing the symmetric convolution of the sequences $\theta(n_1,n_2)$

and $h(n_1,n_2)$ directly. The operators $\mathcal{T}_1\{\bullet\}$, $\mathcal{T}_2\{\bullet\}$, and $\mathcal{T}_3\{\bullet\}$ are different two-dimensional discrete trigonometric transforms. The circular convolution-multiplication property for discrete Fourier transforms, $d(n_1,n_2) = \mathcal{F}^{-1}\{\mathcal{F}\{\theta(n_1,n_2)\}\cdot\mathcal{F}\{h(n_1,n_2)\}\}$, is very similar to the symmetric convolution-multiplication property for discrete trigonometric transforms. The operator $\mathcal{F}\{\bullet\}$ again represents the discrete Fourier transform. The symmetric convolution-multiplication property exists for forty different one-dimensional cases of various combinations of the sixteen transforms in the discrete trigonometric transform family. Symmetric convolution is limited to sequences having underlying symmetry. The forty different cases cover many possibilities for both symmetric and antisymmetric sequences which allows an asymmetric sequence to be decomposed into its symmetric and antisymmetric parts prior to being convolved.

Previous applications of the discrete cosine transform to linear image reconstruction [26], [27] provided very good diagonal approximations for certain types of correlation matrices. Many linear image processing techniques require knowledge of how the individual pixels of an image are correlated with each other. If the elements of the vector $\theta$ represent the pixels of the image $\theta(n_1,n_2)$, the correlation between pixels is expressed by the matrix $\boldsymbol{R}_{\theta\theta} = \overline{\theta\theta^T}$. The overbar, $\bar{}$, indicates the expected value operator, and the superscript '$T$' denotes the transpose of the vector $\theta$. If the size of the image is $N_1 \times N_2$, then $\theta$ will be an $N_1N_2 \times 1$ vector and $\boldsymbol{R}_{\theta\theta}$ will be an $N_1N_2 \times N_1N_2$ matrix. The type-II discrete cosine transform has the property that it very nearly diagonalizes the correlation matrix of an image whose pixels are highly correlated [27]. An image with highly-correlated pixels has few abrupt transitions within a small neighborhood of pixels in the image. The approximate diagonal form of the correlation matrix for a highly-correlated image is $\boldsymbol{R}_{\vartheta_T\vartheta_T} = \boldsymbol{C}_{\mathrm{IIe},N_1N_2}\boldsymbol{R}_{\theta\theta}\boldsymbol{C}^T_{\mathrm{IIe},N_1N_2}$. The $N_1N_2 \times N_1N_2$ matrix $\boldsymbol{C}_{\mathrm{IIe},N_1N_2}$ is a type-II discrete

cosine transform matrix for even-length two-dimensional sequences. The discrete cosine transform of the vector $\theta$ is the vector $\vartheta_T = \boldsymbol{C}_{\mathrm{IIe},N_1N_2}\theta$. The off-diagonal elements of the matrix $\boldsymbol{R}_{\vartheta_T\vartheta_T}$ are all approximately zero. As the pixels in an image become more highly correlated, the off-diagonal elements in its transform-domain correlation matrix, $\boldsymbol{R}_{\vartheta_T\vartheta_T}$, approach zero [39].

One of the results of this research demonstrates another diagonalizing property of discrete trigonometric transforms. Symmetric convolution has an equivalent matrix representation as $\boldsymbol{d} = \boldsymbol{H}_{SC}\theta$. The matrix $\boldsymbol{H}_{SC}$ incorporates all of the symmetric extensions, shifts, additions, and multiplications of symmetric convolution. The resulting vector $\boldsymbol{d}$ contains the same elements as the elements in $d(n_1,n_2)$ which resulted from the symmetric convolution of the two-dimensional sequences $\theta(n_1,n_2)$ and $h(n_1,n_2)$. In the vector-matrix representation, the sequences $\theta(n_1,n_2)$ and $h(n_1,n_2)$ are represented by the vectors $\theta$ and $\boldsymbol{h}$.

Trigonometric transforms diagonalize symmetric convolution matrices because symmetric convolution in the spatial domain is equivalent to point-wise multiplication in the transform domain. The results of this research reveal diagonalizing forms for all forty cases of symmetric convolution. In general, $\boldsymbol{d} = \boldsymbol{T}_3^{-1}\mathcal{H}_T\boldsymbol{T}_1\theta$, where the matrix $\mathcal{H}_T$ is diagonal with the vector $\boldsymbol{T}_2\boldsymbol{h}$ along its diagonal. This expression is exactly equivalent to $\boldsymbol{d} = \boldsymbol{T}_3^{-1}\left[\boldsymbol{T}_1\theta \odot \boldsymbol{T}_2\boldsymbol{h}\right]$, where $\odot$ represents a point-wise or Hadamard product [22]. The three matrices $\boldsymbol{T}_1$, $\boldsymbol{T}_2$, and $\boldsymbol{T}_3$ are matrix representations of two-dimensional trigonometric transforms. The fact that the above matrix equation produces exactly the same result as $d(n_1,n_2) = \mathcal{T}_3^{-1}\left\{\mathcal{T}_1\left\{\theta(n_1,n_2)\right\}\cdot\mathcal{T}_2\left\{h(n_1,n_2)\right\}\right\}$ is one of the main contributions of this research [9].

The matrix form of the symmetric convolution-multiplication property allows for a more natural extension of the property to asymmetric multidimensional sequences than the operator

form of the property. The ability to apply the property to asymmetric two-dimensional sequences is important because they are the most general class of sequences encountered in image reconstruction. The alternate derivation of the matrix form of the property depends on the fact that a discrete Fourier transform matrix diagonalizes a circulant matrix representing circular convolution [23].

The diagonalizing results for symmetric convolution matrices presented here are related to the work of Sánchez *et al.* Her team shows that diagonalizing forms exist for the eight discrete cosine transforms [41] and the eight discrete sine transforms [42]. Their method is to determine the class of matrices whose eigenvectors are sine and cosine transforms, but their results reveal diagonalizing forms for only sixteen of the forty cases of symmetric convolution -- one case of symmetric convolution for each of the sixteen transforms. Their results therefore do not extend to the general case of convolving an asymmetric sequence which requires different convolution relations for each part of the sequence's underlying symmetry. The approach taken here is to relate the discrete trigonometric transform to the discrete Fourier transform which already possesses a diagonalizing form [23]. These newly derived diagonalizing matrix forms then allow for a natural extension of all the cases of the symmetric convolution-multiplication property to multidimensional asymmetric sequences.

The notion that a type-II discrete cosine transform matrix approximately diagonalizes the correlation matrix of a highly-correlated image and the property of the trigonometric transform to exactly diagonalize a symmetric convolution matrix both play important roles in applying trigonometric transforms to image reconstruction problems. Many existing linear image reconstruction techniques rely on knowledge of both the correlation of the pixels in an object being imaged and the impulse response of the system which degrades the object. Optical scientists and engineers refer to the two-dimensional impulse response of the degrading system as its point spread

function. The name arises from the blurring or spreading of individual points comprising the object [16].

As mentioned previously, it is computationally more efficient to calculate convolutional results in the transform domain. The calculations become even more efficient if all the transform domain matrices involved are either diagonal or well-approximated by their diagonal elements. Transform domain implementations of filters which rely on just the diagonal elements of matrices are called *scalar* filters.

Traditional Fourier image reconstruction filters have scalar forms. For an image reconstruction filter to have a scalar form, both the matrix representing convolution of the degrading point spread function and the correlation matrix of the object must have diagonal or approximately diagonal forms. Discrete Fourier transform matrices diagonalize circulant convolution matrices [23]. Discrete Fourier transform matrices also provide approximate diagonal forms of object correlation matrices in the transform domain [48]. For a highly correlated image, the discrete Fourier transform will not, however, condense as much of the content in all the transform domain terms onto the diagonal as a type-II discrete cosine transform [20], [29]. The lower energy in the off-diagonal elements in the cosine transform domain leads to a better diagonal approximation than the discrete Fourier transform.

The results of this research show for the first time that trigonometric transform representations of image reconstruction filters also have scalar forms. The fact that a type-II discrete cosine transform matrix approximately diagonalizes the correlation matrix of a highly-correlated image is well-known [27]. More recently, Martucci's symmetric convolution-multiplication property for discrete trigonometric transforms [34] allows the matrix diagonalization form of symmetric convolution presented here. Thus both matrices underlying the trigonometric image reconstruction filter are either diagonal or approximately diagonal and a scalar filter results.

To help illustrate how a filter can recover an estimate of an object from its distorted, noisy version, consider the imaging scenario in Figure 1. In the figure, $\theta(n_1,n_2)$ represents the original

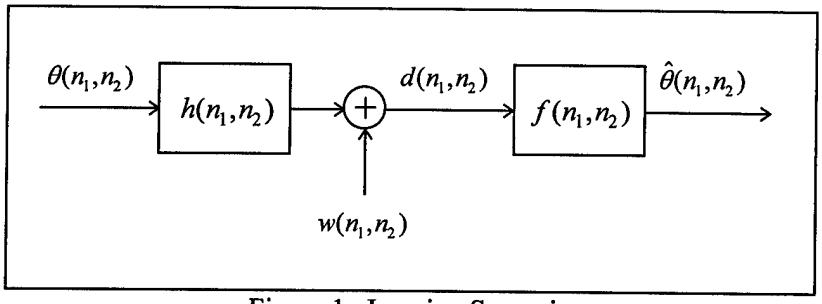



Figure 1. Imaging Scenario

object. The sequence $h(n_1,n_2)$ represents the point spread function of the system blurring the object. The scenario adds noise represented by the sequence $w(n_1,n_2)$ to produce the received data sequence $d(n_1,n_2) = h(n_1,n_2)*\theta(n_1,n_2) + w(n_1,n_2)$. The recovery filter in the scenario has a two-dimensional impulse response, $f(n_1,n_2)$, so that the estimate of the object is $\hat{\theta}(n_1,n_2) = f(n_1,n_2)*d(n_1,n_2)$. The goal of image reconstruction is to find the impulse response of the recovery filter, $f(n_1,n_2)$, which produces the best possible estimate $\hat{\theta}(n_1,n_2)$.

If the noise, $w(n_1,n_2)$, is negligible, then the recovery system can use an inverse filter to produce an estimate $\hat{\theta}(n_1,n_2) = \theta(n_1,n_2)$ which recovers the object exactly. Otherwise an optimum choice for a recovery filter is the Wiener filter. Inverse and Wiener filters are two traditional types of linear image reconstruction filters. They both have scalar representations in the Fourier domain. Inverse filters simply invert the Fourier transform coefficients of the degradation filter to recover the original object [15], [27]. Wiener filters are regularized versions of inverse filters designed to operate in the presence of noise. They minimize the mean-squared error between the estimates they recover and the original object being imaged [21], [38], [44], [52].

This research shows that both inverse and Wiener filters can benefit from the symmetric convolution-multiplication property of discrete trigonometric transforms. The results derived here create new versions of inverse and scalar Wiener filters which exist in the transform domain of the trigonometric transforms.

All scalar Wiener filters, including the traditional Fourier domain and new trigonometric transform domain versions, are approximations of more optimum vector or generalized Wiener filters [38]. A vector Wiener filter accounts for the off-diagonal elements present in the transform domain representation of the object correlation matrix. Vector Wiener filters provide better estimates than scalar Wiener filters [13], [14], but they are more computationally intense than scalar Wiener filters. Pratt [38] shows that the mean-squared error performance of vector Wiener filters in the transform domain is independent of the choice of transform. Different scalar approximations of a vector Wiener filter will, however, yield different levels of performance based on how well the transform domain matrices underlying the filter approximate diagonal matrices. The trigonometric scalar Wiener filter derived here yields better performance than its Fourier counterpart because the discrete cosine transform yields a more accurate diagonal approximation of an object correlation matrix for a highly-correlated image than a discrete Fourier transform [27].

Traditional Fourier-domain scalar Wiener filters are computationally less intense than nonlinear iterative techniques, but suffer from poorer mean-squared error performance. The filters which result from this research require fewer computations than Fourier scalar Wiener filters while demonstrating better mean-squared error performance. They thus provide one possible solution to the Air Force's overall problem of finding image reconstruction techniques which are computationally more efficient than nonlinear techniques, but which have better performance than existing linear methods.

This research effort represents the first time the symmetric convolution-multiplication property of discrete trigonometric transforms [34] has ever been applied to image reconstruction. The focus is therefore on applying the new technique to the most basic image reconstruction methods. To apply this new technique to create new versions of inverse and Wiener filters requires a few basic assumptions which must underlie the model of the basic imaging scenario. These assumptions involve the form of the filters which cause the degradation, the statistics of the object being imaged, and the noise in the model.

The image reconstruction techniques of inverse and Wiener filtering are linear and shift-invariant. The techniques must therefore assume that the degradation is caused by a system which is also linear and shift-invariant. The implication of linearity and shift-invariance is that the form of the point spread function degrading the object is the same for each pixel of the object. This assumption is valid for real-world imaging situations [16]. The trigonometric inverse and Wiener filters derived here require knowledge of the point spread function of the degrading system. Fourier domain inverse and scalar Wiener filters also require a known point spread function. In practice the point spread function of the degrading system is often not known exactly. In many cases it can be modeled as a random process with reasonable assumptions made for its statistics. For example, if the distortion in an image is caused by atmospheric turbulence, then reasonable approximations to a wide variety of atmospheric point spread functions can often be found [40].

An additional limitation within the imaging model for this application of the symmetric convolution-multiplication property is related to the statistics of the object being imaged. The object statistics must be assumed to be wide-sense stationary and highly correlated. A wide-sense stationary object has a constant mean and an autocorrelation which depends only on the difference between the locations of the samples in an image [7]. If the vector $\theta$ represents the

object sequence $\theta(n_1,n_2)$, the mean of the vector $\theta$ is the vector $\mu_\theta$. For a wide-sense stationary object, the vector $\mu_\theta$ must first be constant with the same value for each of its elements. The second condition for the vector $\theta$ to be wide-sense stationary is that the $m$-$n$th element of its correlation matrix, $[\boldsymbol{R}_{\theta\theta}]_{mn}$, must only depend on the difference between $m$ and $n$. An image with highly-correlated pixels will have only minor fluctuations in the values of its pixels in a small region. Both assumptions of wide-sense stationarity and highly-correlated pixels are necessary for the correlation matrix to become almost diagonal in the trigonometric transform domain. A good scalar approximation of the filter will result if the correlation matrix becomes almost diagonal in the transform domain.

The imaging model also includes assumptions regarding the noise in the system. The model requires that the noise samples be zero-mean, additive, of uniform variance, uncorrelated with each other, and independent of the object. These assumptions imply that the amount of noise present in each pixel of the image is completely unrelated to both the object intensity and the amount of noise present in any other pixel. In most imaging situations, this noise model is not as accurate as one which assumes that the noise is an object-dependent Poisson random process [17]. The filtering techniques derived here do not use a more accurate Poisson noise model because this research provides a first look at the benefits of symmetric convolution to image reconstruction. Earlier Fourier domain inverse and Wiener filters use exactly the same noise model as that assumed here. A logical next step for future research would be to extend the theory developed here to incorporate object-dependent photon noise.

The assumptions made for the trigonometric transform domain filters derived here are the very same assumptions underlying traditional Fourier-domain filters. These assumptions allow the performance of the inverse and scalar Wiener filters for trigonometric transforms to be real-

istically compared to their traditional Fourier domain ancestors. The research performed for this dissertation represents the first time the symmetric convolution-multiplication property of trigonometric transforms has ever been applied to image reconstruction of any type, so the comparison to earlier techniques is warranted. The performance results of these brand new trigonometric transform domain filters show that this technique appears to be promising.

This research achieves its overall goal of applying the recently-developed symmetric convolution-multiplication property of the discrete trigonometric transforms [34] to the traditional image reconstruction problems of inverse and Wiener filtering. The results extend the existing theory behind the symmetric convolution-multiplication property by recasting the problem into vector-matrix form and then demonstrating how discrete trigonometric transform matrices diagonalize matrices which represent symmetric convolution. The development then presents vector-matrix forms of the symmetric convolution property for multidimensional asymmetric sequences which represent the most general type of sequences encountered in image reconstruction. The filtering of multidimensional asymmetric sequences is then possible because symmetric convolution is equivalent to multiplication in the transform domain for each of the underlying types of symmetry in an asymmetric image.

The research presented here uses the newly-derived vector-matrix form of symmetric convolution to calculate for the first time inverse and scalar Wiener filters in the trigonometric transform domain. The new forms of the inverse and scalar Wiener filters closely resemble their traditional Fourier domain counterparts. These results demonstrate how a scalar Wiener filter provides better mean-squared error performance for symmetric point spread functions while reducing the required number of computations. The trigonometric transform domain realizations use fewer calculations because the trigonometric transform of a real sequence is also all real. The Fourier transform produces complex sequences in the transform domain.

The remainder of this dissertation is organized as follows. Chapter II provides some background on the discrete Fourier transform and its diagonalizing forms, inverse and Wiener filtering, and the symmetric convolution-multiplication property of discrete trigonometric transforms. The notion of how one and two-dimensional discrete Fourier transform matrices diagonalize circulant and block circulant matrices, respectively, serves as a comparison to similar diagonalizing results derived for the symmetric convolution-multiplication property of trigonometric transforms. The background on inverse and Wiener filtering in the Fourier domain will later be compared to new results for these filters in the trigonometric transform domain. The background on the symmetric convolution-multiplication property of trigonometric transforms is essential to understanding how this property is later extended theoretically and then applied to the inverse and Wiener filtering problems.

All of the material presented in Chapter II summarizes previously conducted work, while the material in Chapters III - V represents the results of new work performed for this dissertation. In Chapter III, the symmetric convolution-multiplication property of discrete trigonometric transforms is derived using vector-matrix methods and then extended to multidimensional and asymmetric sequences. Chapter IV presents the derivation of one and two-dimensional inverse filters expressed in the trigonometric transform domain of symmetrically convolved sequences. Also in Chapter IV are derivations of one and two-dimensional trigonometric transform versions of scalar Wiener filters. Chapter V provides an example of recovering a distorted image using inverse and scalar Wiener filters and details the mean-squared error performance of the scalar Wiener filter. A conclusion appears as Chapter VI.

# II. *Background*

This chapter provides background on the discrete Fourier transform and its diagonalizing forms, inverse and Wiener filtering, and the symmetric convolution-multiplication property of discrete trigonometric transforms. The concept of how one and two-dimensional discrete Fourier transform (DFT) matrices diagonalize circulant and block circulant matrices will later serve as a comparison to similar diagonalizing results for the symmetric convolution-multiplication property of trigonometric transforms. In the second section, inverse and Wiener filters are derived in the Fourier domain. The traditional representation of these filters will be similar to new trigonometric transform versions derived in later chapters. The final section of this chapter introduces the symmetric convolution-multiplication property of trigonometric transforms. This chapter provides a baseline from which the property is later extended to a more general class of signals and then used to derive inverse and Wiener filters in the trigonometric transform domain.

## 2.1 *Diagonalizing Forms of the Discrete Fourier Transform*

The well-known circular convolution-multiplication property of the DFT [37] for one and two dimensions is reviewed in this section. One and two-dimensional DFT matrices are defined and then shown to diagonalize circulant and block circulant matrices, respectively.

Hunt [23] was the first to observe that a DFT matrix diagonalizes a circulant matrix which performs circular convolution. Consider two finite one-dimensional sequences, $h(n)$ and $\theta(n)$, which equal zero outside the interval $0 \le n \le N-1$. The circular convolution of the two sequences is expressible in vector-matrix notation as $\boldsymbol{d} = \boldsymbol{h} \circledcirc \theta = \boldsymbol{H}_C\theta$, where $\circledcirc$ represents circular convolution. The $N \times N$ matrix $\boldsymbol{H}_C$ is circulant as indicated by the subscript '$C$', and has the

vector $\boldsymbol{h}$ as its first column. Hunt proves that $\boldsymbol{H}_C$ is transformed into a diagonal matrix through the relation

$$\boldsymbol{H}_C = \boldsymbol{W}_N \boldsymbol{\mathcal{H}}_F \boldsymbol{W}_N^{-1}. \tag{1}$$

In Eq. (1), $\boldsymbol{W}_N^{-1}$ is the $N \times N$ DFT matrix with $m$-$n$th entry

$$\left[\boldsymbol{W}_N^{-1}\right]_{mn} = \exp\left\{-j\tfrac{2\pi}{N}mn\right\};\ m, n = 0, 1, \ldots, N-1. \tag{2}$$

Its inverse, $\boldsymbol{W}_N$, is the $N \times N$ inverse DFT matrix with $m$-$n$th entry

$$\left[\boldsymbol{W}_N\right]_{mn} = \frac{1}{N}\exp\left\{+j\tfrac{2\pi}{N}mn\right\};\ m, n = 0, 1, \ldots, N-1. \tag{3}$$

The $N \times N$ matrix $\boldsymbol{\mathcal{H}}_F$ in Eq. (1) is diagonal with the elements of the vector $\boldsymbol{W}_N^{-1}\boldsymbol{h}$ along the diagonal. Hunt [23] shows that the elements of $\boldsymbol{W}_N^{-1}\boldsymbol{h}$ are the eigenvalues of $\boldsymbol{H}_C$. The boldface script notation indicates a matrix in the transform domain, while the subscript *'F'* indicates the Fourier transform domain. Thus $\boldsymbol{W}_N^{-1}\boldsymbol{d} = \boldsymbol{\mathcal{H}}_F \boldsymbol{W}_N^{-1}\boldsymbol{\theta}$ or equivalently

$$\boldsymbol{W}_N^{-1}\boldsymbol{d} = \left(\boldsymbol{W}_N^{-1}\boldsymbol{h}\right) \odot \left(\boldsymbol{W}_N^{-1}\boldsymbol{\theta}\right), \tag{4}$$

where $\odot$ represents a point-wise or Hadamard product [22]. Equation (4) is the one-dimensional circular convolution-multiplication property for the DFT expressed in vector-matrix form.

Hunt's results extend naturally to two dimensions [24] using Kronecker (or tensor) products [18]. The Kronecker product, $\otimes$, of an $M_1 \times N_1$ matrix, $\boldsymbol{A}$, and an $M_2 \times N_2$ matrix, $\boldsymbol{B}$, is

$$\boldsymbol{A} \otimes \boldsymbol{B} = \begin{bmatrix} [\boldsymbol{A}]_{00}\boldsymbol{B} & [\boldsymbol{A}]_{01}\boldsymbol{B} & \cdots & [\boldsymbol{A}]_{0(N_1-1)}\boldsymbol{B} \\ [\boldsymbol{A}]_{10}\boldsymbol{B} & [\boldsymbol{A}]_{11}\boldsymbol{B} & \cdots & [\boldsymbol{A}]_{1(N_1-1)}\boldsymbol{B} \\ \vdots & \vdots & \ddots & \vdots \\ [\boldsymbol{A}]_{(M_1-1)0}\boldsymbol{B} & [\boldsymbol{A}]_{(M_1-1)1}\boldsymbol{B} & \cdots & [\boldsymbol{A}]_{(M_1-1)(N_1-1)}\boldsymbol{B} \end{bmatrix}, \tag{5}$$

where $[\boldsymbol{A}]_{mn}$ is the $m$-$n$th entry of the matrix $\boldsymbol{A}$. The Kronecker product $\boldsymbol{A} \otimes \boldsymbol{B}$ has dimension $M_1M_2 \times N_1N_2$. In two dimensions, the lexicographically-ordered vector $\theta = \mathrm{vec}\{\Theta\}$ provides

an alternate means to express an $N_1 \times N_2$ matrix, $\boldsymbol{\Theta}$. The $\text{vec}\{\bullet\}$ operation converts the matrix into an $N_1 N_2 \times 1$ column vector of stacked transposed rows [24]. The vector-matrix expression for the two-dimensional DFT is then

$$\boldsymbol{\vartheta}_F = \left(\boldsymbol{W}_{N_1}^{-1} \otimes \boldsymbol{W}_{N_2}^{-1}\right)\theta. \tag{6}$$

Note that it requires fewer calculations to compute $\boldsymbol{\Phi}_F = \boldsymbol{W}_{N_1}^{-1}\boldsymbol{\Theta}\boldsymbol{W}_{N_2}^{-T}$, where $\boldsymbol{\Phi}_F$ is the matrix representation of the lexicographically-ordered vector, $\boldsymbol{\vartheta}_F$. This matrix form exists because the Fourier transform acts on the rows and columns of the matrix $\boldsymbol{\Theta}$ separately [27]. The connection between the one and two-dimensional DFT is, however, easier to visualize from Eq. (6). Equation (6) is also the only way to calculate the second order moments of the vector $\theta$ in the transform domain when $\theta$ is later modeled as a random vector.

To extend his one-dimensional results to two dimensions, Hunt [24] defines the discrete two-dimensional circular convolution of two $N_1 \times N_2$ matrices, $\boldsymbol{H}$ and $\boldsymbol{\Theta}$, as a vector-matrix operation. The matrices $\boldsymbol{H}$ and $\boldsymbol{\Theta}$ represent finite two-dimensional sequences $h(n_1,n_2)$ and $\theta(n_1,n_2)$ which are zero outside the region $0 \le n_1 \le N_1 - 1,\ 0 \le n_2 \le N_2 - 1$. The vector-matrix form of two-dimensional circular convolution is $\boldsymbol{d} = \boldsymbol{H}_{BC}\theta$, where $\boldsymbol{d}$ and $\theta$ are lexicographically-ordered vectors. The matrix $\boldsymbol{H}_{BC}$ is an $N_1 N_2 \times N_1 N_2$ *block-circulant* matrix, which is a matrix of partitioned blocks of circulant matrices arranged in a circulant pattern. The first column of the matrix $\boldsymbol{H}_{BC}$ will be the vector $\boldsymbol{h} = \text{vec}\{\boldsymbol{H}\}$. Hunt shows that the $N_1 N_2 \times N_1 N_2$ two-dimensional DFT matrix, $\left(\boldsymbol{W}_{N_1}^{-1} \otimes \boldsymbol{W}_{N_2}^{-1}\right)$, diagonalizes $\boldsymbol{H}_{BC}$. Specifically,

$$\boldsymbol{H}_{BC} = \left(\boldsymbol{W}_{N_1} \otimes \boldsymbol{W}_{N_2}\right)\boldsymbol{\mathcal{H}}_F\left(\boldsymbol{W}_{N_1}^{-1} \otimes \boldsymbol{W}_{N_2}^{-1}\right), \tag{7}$$

where $\mathcal{H}_F$ is diagonal with the vector $\left(W_{N_1}^{-1} \otimes W_{N_2}^{-1}\right)\text{vec}\{H\}$ along the diagonal. Equation (7) implies that

$$\left(W_{N_1}^{-1} \otimes W_{N_2}^{-1}\right)d = \left(W_{N_1}^{-1} \otimes W_{N_2}^{-1}\right)h \odot \left(W_{N_1}^{-1} \otimes W_{N_2}^{-1}\right)\theta, \tag{8}$$

which is the two-dimensional version of Eq. (4).

The circular convolution-multiplication property of DFTs greatly reduces the number of calculations in image reconstruction problems which involve linear filtering. The computational cost savings in implementing Eq. (8) instead of $d = H_{BC}\theta$ arises because it requires fewer computations to compute the transforms, point-multiply the results, and then apply an inverse transform than it does to compute the large vector-matrix multiplication directly. Note that the DFT is not a unique transform for implementing filtering operations in the transform domain. This review of the diagonalizing forms for the DFT matrix has established a reference point from which to analyze and later apply the results of more recent developments in the diagonalizing forms of the trigonometric transforms.

## 2.2 *Signal and Image Reconstruction in the Fourier Domain*

The vector-matrix diagonalizing forms of the previous section play an important role in deriving traditional Fourier domain inverse and Wiener filters. These filters can reconstruct one-dimensional signals and two-dimensional images which are degraded by linear processes. This background material on inverse and Wiener filtering in the Fourier domain will later serve as a comparison to similar, but improved versions of these filters derived in the trigonometric transform domain.

**2.2.1** ***Inverse filtering.*** The derivations of inverse filters for one and two dimensions in the Fourier domain are presented in this subsection. The one-dimensional circular convolution-

multiplication property in Eq. (4) is valid for any $N \times 1$ vectors $\boldsymbol{d}$, $\boldsymbol{h}$, and $\theta$, which represent $N$-point sequences $d(n)$, $h(n)$, and $\theta(n)$. All three sequences equal zero outside the interval $0 \le n \le N-1$. In one-dimensional signal reconstruction, the vectors and sequences take on special meaning beyond that of just being arbitrary sequences for which the circular convolution $\boldsymbol{d} = \boldsymbol{H}_C\theta$ holds. The sequence $h(n)$ represents the finite impulse response of a linear shift-invariant filter which distorts a signal $\theta(n)$ to produce a data sequence $d(n)$. The goal of signal reconstruction is to obtain an estimate, $\hat{\theta}(n)$, of $\theta(n)$ from the corrupted data sequence $d(n)$.

A notational variation of this problem is to replace the above sequences with vectors. The goal of the problem is to find the impulse response of a filter, represented by the vector $\boldsymbol{f}$, which recovers the vector $\hat{\theta}$ from $\boldsymbol{d}$, given knowledge of $\boldsymbol{h}$. Because the underlying form of convolution for the DFT is circular, the vector $\boldsymbol{f}$ can be represented as a circulant matrix, $\boldsymbol{F}_C$, so that $\hat{\theta} = \boldsymbol{F}_C\boldsymbol{d} = \boldsymbol{F}_C\boldsymbol{H}_C\theta$. The matrix $\boldsymbol{F}_C$ will recover the vector $\theta$ exactly if $\boldsymbol{F}_C = \boldsymbol{H}_C^{-1}$, or equivalently in the Fourier domain if

$$\boldsymbol{W}_N\mathcal{F}_F\boldsymbol{W}_N^{-1} = \left[\boldsymbol{W}_N\mathcal{H}_F\boldsymbol{W}_N^{-1}\right]^{-1}. \tag{9}$$

It follows that $\mathcal{F}_F = \mathcal{H}_F^{-1}$. Both of the matrices $\mathcal{F}_F$ and $\mathcal{H}_F$ are diagonal, and can be completely characterized by their elements $\mathcal{F}_F(k)$ and $\mathcal{H}_F(k)$, so that Eq. (9) reduces to the scalar equation

$$\mathcal{F}_F(k) = \frac{1}{\mathcal{H}_F(k)}, \tag{10}$$

for $k = 0, 1, \ldots, N-1$, provided $\mathcal{H}_F(k) \neq 0$. Recall that script letters denote transform domain quantities, and the subscript '$F$' denotes the Fourier domain. Equation (10) is the one-dimensional inverse filter expressed in the Fourier domain.

The two-dimensional imaging scenario uses the $N_1N_2 \times 1$ vectors $\boldsymbol{d}$ and $\theta$ to represent a detected image and an original object, respectively. The vectors are lexicographic reorderings of the $N_1 \times N_2$ matrices $\boldsymbol{D}$ and $\boldsymbol{\Theta}$. The two-dimensional finite impulse response, $\boldsymbol{H}$, of a system which distorts an object is its point spread function (PSF). The lexicographically-ordered vector $\boldsymbol{h} = \text{vec}\{\boldsymbol{H}\}$ represents the PSF. As in the previous section, the vector $\boldsymbol{h}$ is the first column of the $N_1N_2 \times N_1N_2$ block circulant matrix $\boldsymbol{H}_{BC}$. In the two-dimensional circular convolution relation, $\boldsymbol{d} = \boldsymbol{H}_{BC}\theta$, the matrix $\boldsymbol{H}_{BC}$ blurs the object to produce the distorted data.

Just as in one dimension, this classical image reconstruction problem is to find the two-dimensional impulse response of a filter which recovers an estimate $\hat{\theta}$ from $\boldsymbol{d}$, given $\boldsymbol{h}$. The impulse response is represented by the lexicographically-ordered vector $\boldsymbol{f}$. The two-dimensional convolution operation of the filter is implemented as a block circulant matrix, $\boldsymbol{F}_{BC}$, with the vector $\boldsymbol{f}$ as its first column. The expression to recover the estimate then becomes $\hat{\theta} = \boldsymbol{F}_{BC}\boldsymbol{d} = \boldsymbol{F}_{BC}\boldsymbol{H}_{BC}\theta$. The object vector $\theta$ is exactly recovered if $\boldsymbol{F}_{BC} = \boldsymbol{H}_{BC}^{-1}$. This problem is commonly presented in the Fourier domain [27], [15] as

$$\left(\boldsymbol{W}_{N_1} \otimes \boldsymbol{W}_{N_2}\right)\mathcal{F}_F\left(\boldsymbol{W}_{N_1}^{-1} \otimes \boldsymbol{W}_{N_2}^{-1}\right) = \left[\left(\boldsymbol{W}_{N_1} \otimes \boldsymbol{W}_{N_2}\right)\mathcal{H}_F\left(\boldsymbol{W}_{N_1}^{-1} \otimes \boldsymbol{W}_{N_2}^{-1}\right)\right]^{-1}. \tag{11}$$

From Eq. (11) it follows that $\mathcal{F}_F = \mathcal{H}_F^{-1}$, where both matrices are diagonal and can be represented by their diagonal elements $\mathcal{F}_F(k_1,k_2)$ and $\mathcal{H}_F(k_1,k_2)$. The scalar form of Eq. (11) is thus

$$\mathcal{F}_F(k_1,k_2) = \frac{1}{\mathcal{H}_F(k_1,k_2)}, \tag{12}$$

for $k_1 = 0,\ 1,\ \ldots, N_1 - 1$; $k_2 = 0,\ 1,\ \ldots, N_2 - 1$; provided $\mathcal{H}_F(k_1,k_2) \neq 0$. Equation (12) is the two-dimensional Fourier domain inverse filter. Note its similarity to Eq. (10).

One limitation of inverse filtering in the Fourier domain [27], [15] is that for most practical realizations, $\mathcal{H}_F(k_1,k_2)$ takes on very small values for increasing $k_1$ and $k_2$. These small values produce very high gains in $\mathcal{T}_F(k_1,k_2)$ at the higher spatial frequencies in Eq. (12). This problem becomes quite serious when the image model expands to incorporate noise with a uniform power spectral density across all frequencies. The solution to this problem involves regularizing the high frequency gain of the inverse filter.

**2.2.2** ***Wiener filtering***. The development of scalar Wiener filters for one and two dimensions in the Fourier transform domain is reviewed in this subsection. Wiener filters bear the name of Norbert Wiener who pioneered them in the 1940s during studies on time series analysis [52]. Helstrom [21] and Slepian [44] later applied Wiener filters to the image reconstruction problem. These filters introduce a degree of regularization to the inverse problem by minimizing the mean-squared error between the object estimate they produce and the original object. They have better mean-squared error performance than inverse filters when reconstructing noisy signals or images.

The data model of the previous sections must have an added term to incorporate noise so that the model now becomes $\boldsymbol{d} = \boldsymbol{H}_C\theta + \boldsymbol{w}$ for the one-dimensional circulant case. In the new model, $\boldsymbol{w}$ is an $N \times 1$ zero-mean uniform-variance noise vector whose samples are uncorrelated both with the object vector, $\theta$, and with each other. The model assumes that the object vector, $\theta$, is also a random vector which will, in general, have a mean vector $\mu_\theta = E\{\theta\} = \overline{\theta}$. The operator $E\{\bullet\}$ and the overbar, $\overline{\bullet}$, are equivalent ways to denote expectation. The vector $\theta$ will also have a covariance matrix $\boldsymbol{C}_{\theta\theta} = E\left\{(\theta - \mu_\theta)(\theta - \mu_\theta)^T\right\}$. Under these assumptions, Wiener filters produce an estimate represented by the vector $\hat{\theta}$ which minimizes the expected value of

the square of the 2-norm of the error vector, $\varepsilon = \theta - \hat{\theta}$. If $\|\varepsilon\|_2$ denotes the 2-norm of the vector $\varepsilon$, then the linear minimum mean-squared error estimate, $\hat{\theta}_{\min}$, occurs at $\min\left\{\overline{\|\varepsilon\|_2^2}\right\}$ or equivalently $\min\left\{\overline{\varepsilon^T\varepsilon}\right\}$.

Solutions to the problem of finding filters which recover $\hat{\theta}$ from $\boldsymbol{d}$ under these conditions appear in many statistical signal processing texts [28], [43], [48]. The solution which follows is from [28] and is the linear minimum mean-squared error estimate which results from the Bayesian Gauss-Markov Theorem under the conditions described above:

$$\hat{\theta} = \mu_\theta + \boldsymbol{C}_{\theta\theta}\boldsymbol{H}_C^T\left[\boldsymbol{H}_C\boldsymbol{C}_{\theta\theta}\boldsymbol{H}_C^T + \boldsymbol{C}_{ww}\right]^{-1}\left(\boldsymbol{d} - \boldsymbol{H}_C\mu_\theta\right). \tag{13}$$

If the mean of the vector $\theta$ is constant, then without loss of generality [28], the mean can be assumed to be the zero vector, $\mathbf{0}$, so that Eq. (13) becomes

$$\hat{\theta} = \boldsymbol{R}_{\theta\theta}\boldsymbol{H}_C^T\left[\boldsymbol{H}_C\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_C^T + \boldsymbol{R}_{ww}\right]^{-1}\boldsymbol{d}. \tag{14}$$

In Eq. (14) the covariance matrices of Eq. (13) are replaced with correlation matrices defined by $\boldsymbol{R}_{\theta\theta} = \overline{\theta\theta^T}$. In this case, $\boldsymbol{C}_{\theta\theta}$ will equal $\boldsymbol{R}_{\theta\theta}$ because of the assumption that the vector $\mu_\theta = \mathbf{0}$.

It is not difficult to derive Eq. (14) using standard vector-matrix minimization techniques as outlined in [38]. First observe that the goal of the problem is to find the filter which minimizes $\overline{\|\varepsilon\|_2^2} = \overline{\varepsilon^T\varepsilon} = \mathrm{Tr}\left\{\overline{\varepsilon^T\varepsilon}\right\} = \mathrm{Tr}\left\{\overline{\varepsilon\varepsilon^T}\right\}$. The operation $\mathrm{Tr}\{\bullet\}$ denotes the trace of a matrix. The transpose denoted by the superscript '$T$' is used because it is assumed that all sequence domain quantities are real. Substituting $\theta - \hat{\theta}$ for the error vector, $\varepsilon$, produces

$$\begin{aligned}\overline{\|\varepsilon\|_2^2} &= \mathrm{Tr}\left\{E\left\{\left(\theta-\hat{\theta}\right)\left(\theta-\hat{\theta}\right)^T\right\}\right\} \\ &= \mathrm{Tr}\left\{E\left\{\theta\theta^T - \theta\hat{\theta}^T - \hat{\theta}\theta^T + \hat{\theta}\hat{\theta}^T\right\}\right\} \ldots\end{aligned}$$

$$= \mathrm{Tr}\left\{E\left\{\theta\theta^T - 2\hat{\theta}\theta^T + \hat{\theta}\hat{\theta}^T\right\}\right\}. \tag{15}$$

The intermediate steps of Eq. (15) use the fact that for any two real vectors, $\mathrm{Tr}\{\boldsymbol{ab}^T\} = \mathrm{Tr}\{\boldsymbol{b}^T\boldsymbol{a}\} = \boldsymbol{b}^T\boldsymbol{a} = \boldsymbol{a}^T\boldsymbol{b} = \mathrm{Tr}\{\boldsymbol{a}^T\boldsymbol{b}\} = \mathrm{Tr}\{\boldsymbol{ba}^T\}$. If $\boldsymbol{F}_C$ is the circulant matrix version of the one-dimensional impulse response of the reconstruction filter, then substituting for $\hat{\theta} = \boldsymbol{F}_C\boldsymbol{d} = \boldsymbol{F}_C\boldsymbol{H}_C\theta + \boldsymbol{F}_C\boldsymbol{w}$ yields

$$\overline{\|\varepsilon\|_2^2} = \mathrm{Tr}\left\{E\left\{\theta\theta^T - 2\left(\boldsymbol{F}_C\boldsymbol{H}_C\theta + \boldsymbol{F}_C\boldsymbol{w}\right)\theta^T + \left(\boldsymbol{F}_C\boldsymbol{H}_C\theta + \boldsymbol{F}_C\boldsymbol{w}\right)\left(\boldsymbol{F}_C\boldsymbol{H}_C\theta + \boldsymbol{F}_C\boldsymbol{w}\right)^T\right\}\right\}. \tag{16}$$

After expanding and recognizing that because the object and noise are uncorrelated, $\overline{\boldsymbol{w}\theta^T} = \overline{\theta\boldsymbol{w}^T} = \boldsymbol{0}$, an $N \times N$ zero matrix, Eq. (16) becomes

$$\overline{\|\varepsilon\|_2^2} = \mathrm{Tr}\left\{\boldsymbol{R}_{\theta\theta} - 2\boldsymbol{F}_C\boldsymbol{H}_C\boldsymbol{R}_{\theta\theta} + \boldsymbol{F}_C\left(\boldsymbol{H}_C\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_C^T + \boldsymbol{R}_{ww}\right)\boldsymbol{F}_C^T\right\}, \tag{17}$$

where $\boldsymbol{R}_{\theta\theta} = \overline{\theta\theta^T}$ and $\boldsymbol{R}_{ww} = \overline{\boldsymbol{w}\boldsymbol{w}^T}$. The next step in finding the filter, $\boldsymbol{F}_C$, which minimizes $\overline{\|\varepsilon\|_2^2}$, is to take the first derivative of Eq. (17) with respect to $\boldsymbol{F}_C$ which yields

$$\frac{\partial\overline{\|\varepsilon\|_2^2}}{\partial\boldsymbol{F}_C} = -2\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_C^T + 2\boldsymbol{F}_C\left(\boldsymbol{H}_C\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_C^T + \boldsymbol{R}_{ww}\right). \tag{18}$$

Taking the derivative of Eq. (17) requires using the matrix derivatives $\partial\mathrm{Tr}\{\boldsymbol{XA}\}/\partial\boldsymbol{X} = \boldsymbol{A}^T$ and $\partial\mathrm{Tr}\{\boldsymbol{XAX}^T\}/\partial\boldsymbol{X} = \boldsymbol{X}\left(\boldsymbol{A} + \boldsymbol{A}^T\right) = 2\boldsymbol{XA}$, if $\boldsymbol{A}$ is symmetric [54].

Setting Eq. (18) equal to the zero matrix and solving for $\boldsymbol{F}_C$ produces the filter which minimizes the mean-squared error between the original object and its estimate,

$$\boldsymbol{F}_C = \boldsymbol{R}_{\theta\theta}\boldsymbol{H}_C^T\left[\boldsymbol{H}_C\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_C^T + \boldsymbol{R}_{ww}\right]^{-1}. \tag{19}$$

Equation (19) is exactly equivalent to Eq. (14) because $\hat{\theta} = \boldsymbol{F}_C \boldsymbol{d}$. Converting Eq. (19) to the Fourier domain using Eq. (1), produces

$$\boldsymbol{W}_N \boldsymbol{\mathcal{F}}_F \boldsymbol{W}_N^{-1} = \boldsymbol{R}_{\theta\theta}\left(\boldsymbol{W}_N \boldsymbol{\mathcal{H}}_F \boldsymbol{W}_N^{-1}\right)^H \left[\left(\boldsymbol{W}_N \boldsymbol{\mathcal{H}}_F \boldsymbol{W}_N^{-1}\right)\boldsymbol{R}_{\theta\theta}\left(\boldsymbol{W}_N \boldsymbol{\mathcal{H}}_F \boldsymbol{W}_N^{-1}\right)^H + \boldsymbol{R}_{ww}\right]^{-1}. \tag{20}$$

The superscript *'H'* which denotes the Hermitian or conjugate transpose is now required because Fourier domain quantities are complex. Solving Eq. (20) for $\boldsymbol{\mathcal{F}}_F$ and distributing the Hermitian operator results in

$$\begin{aligned} \boldsymbol{\mathcal{F}}_F &= \boldsymbol{W}_N^{-1}\boldsymbol{R}_{\theta\theta}\boldsymbol{W}_N^{-H}\boldsymbol{\mathcal{H}}_F^H \boldsymbol{W}_N^H \left[\boldsymbol{W}_N \boldsymbol{\mathcal{H}}_F \boldsymbol{W}_N^{-1}\boldsymbol{R}_{\theta\theta}\boldsymbol{W}_N^{-H}\boldsymbol{\mathcal{H}}_F^H \boldsymbol{W}_N^H + \boldsymbol{R}_{ww}\right]^{-1}\boldsymbol{W}_N \\ &= \boldsymbol{W}_N^{-1}\boldsymbol{R}_{\theta\theta}\boldsymbol{W}_N^{-H}\boldsymbol{\mathcal{H}}_F^* \left[\boldsymbol{W}_N^{-1}\boldsymbol{W}_N \boldsymbol{\mathcal{H}}_F \boldsymbol{W}_N^{-1}\boldsymbol{R}_{\theta\theta}\boldsymbol{W}_N^{-H}\boldsymbol{\mathcal{H}}_F^* \boldsymbol{W}_N^H \boldsymbol{W}_N^{-H} + \boldsymbol{W}_N^{-1}\boldsymbol{R}_{ww}\boldsymbol{W}_N^{-H}\right]^{-1}, \end{aligned} \tag{21}$$

where $\boldsymbol{\mathcal{H}}_F^H = \boldsymbol{\mathcal{H}}_F^*$ because $\boldsymbol{\mathcal{H}}_F$ is diagonal. Equation (21) reduces to

$$\boldsymbol{\mathcal{F}}_F = \boldsymbol{R}_{\Theta_F\Theta_F}\boldsymbol{\mathcal{H}}_F^*\left[\boldsymbol{\mathcal{H}}_F \boldsymbol{R}_{\Theta_F\Theta_F}\boldsymbol{\mathcal{H}}_F^* + \boldsymbol{R}_{w_F w_F}\right]^{-1}, \tag{22}$$

where $\boldsymbol{R}_{\Theta_F\Theta_F} = \boldsymbol{W}_N^{-1}\boldsymbol{R}_{\theta\theta}\boldsymbol{W}_N^{-H}$ and $\boldsymbol{R}_{w_F w_F} = \boldsymbol{W}_N^{-1}\boldsymbol{R}_{ww}\boldsymbol{W}_N^{-H}$. Equation (22) is the *general* or *vector Wiener filter* in the Fourier transform domain [38].

Note that in Eq. (22) the matrices $\boldsymbol{\mathcal{F}}_F$, $\boldsymbol{\mathcal{H}}_F$, and $\boldsymbol{\mathcal{H}}_F^*$ are diagonal by definition, and the matrix $\boldsymbol{R}_{w_F w_F}$ is diagonal because of the assumption of uniform-variance uncorrelated noise samples. The matrix $\boldsymbol{R}_{\Theta_F\Theta_F}$ will be approximately diagonal under the assumption of wide-sense stationarity for the object. The approximation arises because the discrete Fourier transform does not exactly diagonalize the symmetric Toeplitz form of the correlation matrix for a wide-sense stationary object [48]. The scalar equation which results from approximating $\boldsymbol{R}_{\Theta_F\Theta_F}$ by retaining only its diagonal elements is then

$$\mathcal{F}_F(k) = \frac{\mathcal{H}_F^*(k)}{\left|\mathcal{H}_F(k)\right|^2 + \dfrac{\overline{\mathcal{W}_F^2(k)}}{\overline{\Theta_F^2(k)}}}. \tag{23}$$

The terms $\overline{\mathcal{W}_F^2(k)}$ and $\overline{\Theta_F^2(k)}$ are the diagonal elements of $\boldsymbol{R}_{\mathcal{W}_F\mathcal{W}_F}$ and $\boldsymbol{R}_{\Theta_F\Theta_F}$ and represent the power spectral densities of the noise and object respectively. The terms $\mathcal{F}_F(k)$ and $\mathcal{H}_F(k)$ are the elements of the diagonal matrices $\mathcal{F}_F$ and $\mathcal{H}_F$. Equation (23) also uses the fact that $\mathcal{H}_F(k)\mathcal{H}_F^*(k) = \left|\mathcal{H}_F(k)\right|^2$. The Fourier domain filter in Eq. (23) is referred to as the one-dimensional *scalar Wiener filter*. Under the assumption of wide-sense stationarity for the object, it is a good approximation to the vector Wiener filter [38].

The extension of the preceding development to two dimensions is straightforward. The data model incorporating noise becomes $\boldsymbol{d} = \boldsymbol{H}_{BC}\theta + \boldsymbol{w}$. The $N_1N_2 \times N_1N_2$ block circulant degradation matrix $\boldsymbol{H}_{BC}$ has the lexicographically-ordered vector $\boldsymbol{h} = \text{vec}\{\boldsymbol{H}\}$ which represents the PSF as its first column. The vectors $\boldsymbol{d}$ and $\theta$ are $N_1N_2 \times 1$ lexicographically-ordered vectors representing the data and the original object matrices $\boldsymbol{D}$ and $\Theta$ respectively. The lexicographically-ordered vector $\boldsymbol{w}$ is an $N_1N_2 \times 1$ zero-mean uniform-variance noise vector whose samples are again assumed uncorrelated both with the object vector, $\theta$, and with each other. The object vector, $\theta$, is again random. It has a constant mean vector $\mu_\theta$, which equals $\mathbf{0}$ without loss of generality. The autocorrelation matrix of the vector $\theta$ is $\boldsymbol{R}_{\theta\theta} = \overline{\theta\theta^T}$. The linear minimum mean-squared error estimate represented by the vector $\hat{\theta}$ is then [28]

$$\hat{\theta} = \boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{BC}^T\left[\boldsymbol{H}_{BC}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{BC}^T + \boldsymbol{R}_{ww}\right]^{-1}\boldsymbol{d}. \tag{24}$$

The only differences between Eqs. (14) and (24) are that in the two-dimensional case the degradation matrix, $\boldsymbol{H}_{BC}$, is block circulant, the autocorrelation matrix, $\boldsymbol{R}_{\theta\theta}$, is block symmetric

Toeplitz, and the vectors $\hat{\theta}$, $\theta$, $h$, $w$, and $d$ are all lexicographically-ordered. From Eq. (24) it follows that the two-dimensional impulse response of the reconstruction filter is

$$F_{BC} = R_{\theta\theta} H_{BC}^T \left[ H_{BC} R_{\theta\theta} H_{BC}^T + R_{ww} \right]^{-1}. \tag{25}$$

Expanding Eq. (25) using Eq. (7) produces

$$\mathcal{F}_F = R_{\Theta_F \Theta_F} \mathcal{H}_F^* \left[ \mathcal{H}_F R_{\Theta_F \Theta_F} \mathcal{H}_F^* + R_{\mathcal{W}_F \mathcal{W}_F} \right]^{-1}, \tag{26}$$

in the Fourier domain. As in the one-dimensional case, the vector Wiener filter of Eq. (26) is well-approximated by the scalar equation [15]

$$\mathcal{F}_F(k_1,k_2) = \frac{\mathcal{H}_F^*(k_1,k_2)}{\left|\mathcal{H}_F(k_1,k_2)\right|^2 + \dfrac{\overline{\mathcal{W}_F^2(k_1,k_2)}}{\overline{\Theta_F^2(k_1,k_2)}}}, \tag{27}$$

which retains only the diagonal elements of the matrices in Eq. (26). The terms $\overline{\mathcal{W}_F^2(k_1,k_2)}$ and $\overline{\Theta_F^2(k_1,k_2)}$ are the diagonal elements of $R_{\mathcal{W}_F \mathcal{W}_F}$ and $R_{\Theta_F \Theta_F}$ which represent the two-dimensional power spectral densities of the noise and object respectively. The Fourier domain filter in Eq. (27) is the two-dimensional scalar Wiener filter for reconstructing a corrupted image in noise given knowledge of the degrading PSF which distorted it.

The overall goal of this research is to derive new expressions for the one and two-dimensional inverse and scalar Wiener filters of Eqs. (10), (12), (23), and (27) which lie not in the Fourier domain, but which instead lie in the trigonometric transform domain. These filters will also employ the newly-developed symmetric convolution-multiplication property of trigonometric transforms [34]. To derive expressions for these filters in the trigonometric transform domain which employ this new property, it is necessary to first provide some background on the symmetric convolution-multiplication property itself.

## 2.3 *Symmetric Convolution and the Discrete Trigonometric Transforms*

This section provides background on the recently-developed symmetric convolution-multiplication property of discrete trigonometric transforms (DTTs) [34]. The material presented here serves as an essential baseline from which the property will be theoretically extended to a broader class of signals in later chapters and then applied to traditional inverse and Wiener filtering problems.

The discrete cosine transform was first introduced in 1974 [1]. Since then it has been expanded into an entire family of trigonometric transforms [26], [39] consisting of sixteen DTTs which are even and odd-length versions of the discrete sine and cosine transforms (DSTs and DCTs). The family lies at the heart of many image transform coding applications [27], [33], [39]. Although useful for transform coding, the DTTs have not proved very useful in a wide variety of image filtering applications, because until recently no convolution-multiplication property existed for the entire family.

Martucci [34] has recently developed a convolution-multiplication property for the family of DTTs. He defines *symmetric convolution* as the form of convolution for DTTs. His results are analogous to the circular convolution-multiplication property of the DFT [37]. *The symmetric convolution-multiplication property states that an inverse trigonometric transform of the product of the trigonometric transforms of two sequences yields the same result as the symmetric convolution of the two sequences* [34]. The symmetric convolution-multiplication property exists for forty different combinations of the sixteen transforms in the DTT family based on the underlying symmetric periodicities of the different DTTs.

There are four ways to symmetrically extend a finite sequence about a single point of symmetry. These are *whole-sample symmetry* (WS), *whole-sample antisymmetry* (WA), *half-*

*sample symmetry* (HS), and *half-sample antisymmetry* (HA). An example of each appears in Figure 2. Note that the point of symmetry for the WS sequence is the end point in the finite se-

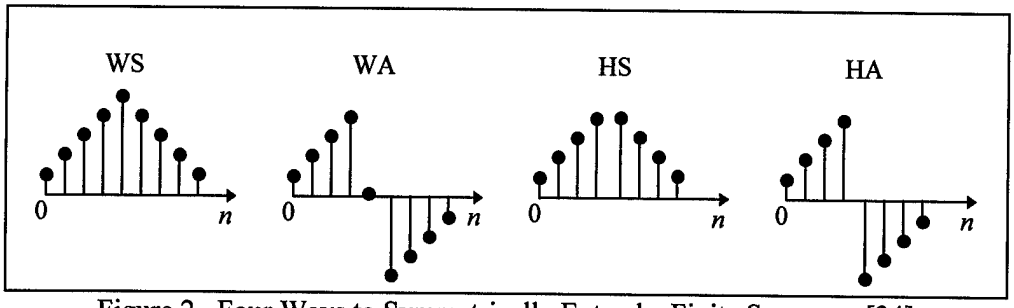


Figure 2. Four Ways to Symmetrically Extend a Finite Sequence [34]

quence before extension, and the point of symmetry for the WA sequence is a zero which must appear after the end point before extension. The points of symmetry for both the HS and HA sequences lie one-half sample beyond the end points in each finite sequence before extension.

There are 16 *symmetric periodic sequences* (SPS's) which result from symmetrically extending a finite sequence to the left using one of the four ways and to the right using possibly a different way. A convention for naming each of the 16 SPS's is to label first the left symmetric extension and then the right symmetric extension. For example a WSHA sequence would exhibit whole-sample symmetry to the left and half-sample antisymmetry to the right.

A finite sequence is converted into an SPS by applying a symmetric extension operator. For example, the WSHA extension of $x(n)$, for $n = 0,\ 1,\ \ldots,\ N-1,$ is

$$\tilde{x}(n) = \varepsilon_{WSHA}\{x(n)\} = \begin{cases} x(n), & n = 0,\ 1,\ \ldots,\ N-1 \\ -x(M-n), & n = N,\ \ldots,\ M-1, \end{cases} \tag{28}$$

where $M = 2N-1$. The symmetric extension operator $\varepsilon_{WSHA}\{\bullet\}$ is one of sixteen different symmetric extension operators -- one for each type of SPS. If $\boldsymbol{x}$ is a column vector whose indi-

vidual entries are $x(n)$, then Eq. (28) is equivalent to the vector-matrix equation, $\tilde{\boldsymbol{x}} = \boldsymbol{E}_{WSHA}\boldsymbol{x}$, where $\tilde{\boldsymbol{x}}$ is an $M \times 1$ column vector. The $M \times N$ matrix $\boldsymbol{E}_{WSHA}$ is defined by

$$\tilde{\boldsymbol{x}} = \boldsymbol{E}_{WSHA}\boldsymbol{x} = \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ \vdots \\ x(N-1) \\ -x(N-1) \\ \vdots \\ -x(2) \\ -x(1) \end{bmatrix} = \begin{bmatrix} 1 & & & & & \\ & 1 & & & & \\ & & 1 & & & \\ & & & \ddots & & \\ & & & & 1 & \\ & & & & & -1 \\ & & & & \iddots & \\ & & & -1 & & \\ 0 & -1 & & & & \end{bmatrix} \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ \vdots \\ x(N-1) \end{bmatrix}. \tag{29}$$

The definitions of the remaining symmetric extension operators follow in a similar fashion.

Each of the 16 DTTs has an underlying symmetry for both the input sequence and the output sequence in the trigonometric transform domain. This behavior is similar to the underlying periodicity of the input and output sequences for the DFT. The underlying input and output symmetries for each DTT as well as its form and other information appears in Table 1. The subscripts on each transform denote the type (from types I - IV) and whether the transform is even or odd. Note that unlike the previous notational convention, no subscript, '*N*,' indicating the length of the transform appears in the table. Omitting the subscript preserves space in the table, and this information appears in the columns for the input and output ranges. Also the constant $k_n$ which appears in the definition of the transforms should not be confused with the one-dimensional transform domain index, $k$, or with the two-dimensional transform domain indices, $k_1$ or $k_2$.

The form of each transform in Table 1 is slightly different from traditional representations of the DTTs [26], [39], [50]. Martucci [34] refers to the form in Table 1 as the *convolution form* of the DTTs and relates each transform to its representation in terms of the *generalized discrete*

Table 1. Forward Discrete Trigonometric Transforms

| Type | *m-n*th Entry | Input | | Output | | Equivalent Transform |
|---|---|---|---|---|---|---|
| | | Symmetry | Range | Symmetry | Range | |
| $\mathcal{C}_{1e}$ | $[C_{1e}]_{mn} = 2k_n \cos\left(\frac{\pi mn}{N}\right)$ | WSWS | $0 \rightarrow N$ | WSWS | $0 \rightarrow N$ | $\mathcal{G}_{0,0}\mathcal{E}_{WSWS}$ |
| $\mathcal{S}_{1e}$ | $[S_{1e}]_{mn} = 2\sin\left(\frac{\pi mn}{N}\right)$ | WAWA | $1 \rightarrow N\text{-}1$ | WAWA | $1 \rightarrow N\text{-}1$ | $j\mathcal{G}_{0,0}\mathcal{E}_{WAWA}$ |
| $\mathcal{C}_{2e}$ | $[C_{2e}]_{mn} = 2\cos\left(\frac{\pi m(n+\frac{1}{2})}{N}\right)$ | HSHS | $0 \rightarrow N\text{-}1$ | WSWA | $0 \rightarrow N\text{-}1$ | $\mathcal{G}_{0,\frac{1}{2}}\mathcal{E}_{HSHS}$ |
| $\mathcal{S}_{2e}$ | $[S_{2e}]_{mn} = 2\sin\left(\frac{\pi m(n+\frac{1}{2})}{N}\right)$ | HAHA | $0 \rightarrow N\text{-}1$ | WAWS | $1 \rightarrow N$ | $j\mathcal{G}_{0,\frac{1}{2}}\mathcal{E}_{HAHA}$ |
| $\mathcal{C}_{3e}$ | $[C_{3e}]_{mn} = 2k_n \cos\left(\frac{\pi(m+\frac{1}{2})n}{N}\right)$ | WSWA | $0 \rightarrow N\text{-}1$ | HSHS | $0 \rightarrow N\text{-}1$ | $\mathcal{G}_{\frac{1}{2},0}\mathcal{E}_{WSWA}$ |
| $\mathcal{S}_{3e}$ | $[S_{3e}]_{mn} = 2k_n \sin\left(\frac{\pi(m+\frac{1}{2})n}{N}\right)$ | WAWS | $1 \rightarrow N$ | HAHA | $0 \rightarrow N\text{-}1$ | $j\mathcal{G}_{\frac{1}{2},0}\mathcal{E}_{WAWS}$ |
| $\mathcal{C}_{4e}$ | $[C_{4e}]_{mn} = 2\cos\left(\frac{\pi(m+\frac{1}{2})(n+\frac{1}{2})}{N}\right)$ | HSHA | $0 \rightarrow N\text{-}1$ | HSHA | $0 \rightarrow N\text{-}1$ | $\mathcal{G}_{\frac{1}{2},\frac{1}{2}}\mathcal{E}_{HSHA}$ |
| $\mathcal{S}_{4e}$ | $[S_{4e}]_{mn} = 2\sin\left(\frac{\pi(m+\frac{1}{2})(n+\frac{1}{2})}{N}\right)$ | HAHS | $0 \rightarrow N\text{-}1$ | HAHS | $0 \rightarrow N\text{-}1$ | $j\mathcal{G}_{\frac{1}{2},\frac{1}{2}}\mathcal{E}_{HAHS}$ |
| $\mathcal{C}_{1o}$ | $[C_{1o}]_{mn} = 2k_n \cos\left(\frac{2\pi mn}{2N-1}\right)$ | WSHS | $0 \rightarrow N\text{-}1$ | WSHS | $0 \rightarrow N\text{-}1$ | $\mathcal{G}_{0,0}\mathcal{E}_{WSHS}$ |
| $\mathcal{S}_{1o}$ | $[S_{1o}]_{mn} = 2\sin\left(\frac{2\pi mn}{2N-1}\right)$ | WAHA | $1 \rightarrow N\text{-}1$ | WAHA | $1 \rightarrow N\text{-}1$ | $j\mathcal{G}_{0,0}\mathcal{E}_{WAHA}$ |
| $\mathcal{C}_{2o}$ | $[C_{2o}]_{mn} = 2l_n \cos\left(\frac{2\pi m(n+\frac{1}{2})}{2N-1}\right)$ | HSWS | $0 \rightarrow N\text{-}1$ | WSHA | $0 \rightarrow N\text{-}1$ | $\mathcal{G}_{0,\frac{1}{2}}\mathcal{E}_{HSWS}$ |
| $\mathcal{S}_{2o}$ | $[S_{2o}]_{mn} = 2\sin\left(\frac{2\pi m(n+\frac{1}{2})}{2N-1}\right)$ | HAWA | $0 \rightarrow N\text{-}2$ | WAHS | $1 \rightarrow N\text{-}1$ | $j\mathcal{G}_{0,\frac{1}{2}}\mathcal{E}_{HAWA}$ |
| $\mathcal{C}_{3o}$ | $[C_{3o}]_{mn} = 2k_n \cos\left(\frac{2\pi(m+\frac{1}{2})n}{2N-1}\right)$ | WSHA | $0 \rightarrow N\text{-}1$ | HSWS | $0 \rightarrow N\text{-}1$ | $\mathcal{G}_{\frac{1}{2},0}\mathcal{E}_{WSHA}$ |
| $\mathcal{S}_{3o}$ | $[S_{3o}]_{mn} = 2\sin\left(\frac{2\pi(m+\frac{1}{2})n}{2N-1}\right)$ | WAHS | $1 \rightarrow N\text{-}1$ | HAWA | $0 \rightarrow N\text{-}2$ | $j\mathcal{G}_{\frac{1}{2},0}\mathcal{E}_{WAHS}$ |
| $\mathcal{C}_{4o}$ | $[C_{4o}]_{mn} = 2\cos\left(\frac{2\pi(m+\frac{1}{2})(n+\frac{1}{2})}{2N-1}\right)$ | HSWA | $0 \rightarrow N\text{-}2$ | HSWA | $0 \rightarrow N\text{-}2$ | $\mathcal{G}_{\frac{1}{2},\frac{1}{2}}\mathcal{E}_{HSWA}$ |
| $\mathcal{S}_{4o}$ | $[S_{4o}]_{mn} = 2l_n \sin\left(\frac{2\pi(m+\frac{1}{2})(n+\frac{1}{2})}{2N-1}\right)$ | HAWS | $0 \rightarrow N\text{-}1$ | HAWS | $0 \rightarrow N\text{-}1$ | $j\mathcal{G}_{\frac{1}{2},\frac{1}{2}}\mathcal{E}_{HAWS}$ |

$$k_n = \begin{cases} 1/2, & n = 0 \text{ or } N \\ 1, & n = 1, 2, \ldots, N-1 \end{cases} \qquad l_n = \begin{cases} 1, & n = 0, 1, \ldots, N-2 \\ 1/2, & n = N-1 \end{cases}$$

*Fourier transform* (GDFT) [3]. The GDFT operator, represented by the script character, $\mathcal{G}_{a,b}\{\bullet\}$, is expressible as a matrix with *m-n*th entry

$$\left[\boldsymbol{G}_{a,b,M}\right]_{mn} = \exp\left(-j\frac{2\pi}{M}(m+a)(n+b)\right); \quad m, n = 0, 1, \ldots, M-1. \tag{30}$$

The boldface character $\boldsymbol{G}_{a,b,M}$ in Eq. (30) denotes the matrix which represents the operator, $\mathcal{G}_{a,b}\{\bullet\}$. The subscripts *'a'* and *'b'* indicate the amounts of the shifts in the sequence and transform domains. The subscript *'M'* denotes the $M \times M$ dimension of the transform matrix which results from extending the original sequence length so that $M = 2N$ for $N$ even or $M = 2N-1$ for $N$ odd. Expressed in this form, each transform in Table 1 is equivalent to the GDFT of a symmetrically extended version of the finite input sequence with the same symmetry that underlies the trigonometric transform. For example, $\mathcal{C}_{1e}\{h(n)\} = \mathcal{G}_{0,0}\{\varepsilon_{WSWS}\{h(n)\}\}$, where $h(n)$ is a finite sequence which the type-I DCT assumes has WSWS underlying symmetry. These equivalent relationships also appear in the last column of Table 1 for each transform. The values of $a$ and $b$ in the GDFT for these cases are always either 0 or $\frac{1}{2}$.

Table 2 lists similar information to Table 1 for each inverse DTT. The second column of Table 2 lists the relationship between the forward and inverse transforms.

Based on the underlying principles discussed above, Martucci [34] defines the symmetric convolution of two sequences, $\theta(n)$ and $h(n)$, as

$$d(n) = \mathcal{R}_L(n)\left[\varepsilon_1\{\theta(n)\} \circledast \varepsilon_2\{h(n)\}\right], \tag{31}$$

where $\mathcal{R}_L(n)$ is a rectangular window of length $L$ which extracts the representative samples from the result. The operations $\varepsilon_1\{\theta(n)\}$ and $\varepsilon_2\{h(n)\}$ create symmetric periodic extensions of the two sequences being convolved. The convolution operation '$\circledast$' represents either circular or skew-circular convolution. The need to perform skew-circular convolution arises because in half of the forty cases of symmetric convolution, the two SPS's to be convolved are not strictly periodic, but are actually anti-periodic depending on the symmetric extension operator for that case.

Table 2. Inverse Discrete Trigonometric Transforms

| Type | Relation to Forward Matrix | Input | | Output | | Equivalent Transform |
|---|---|---|---|---|---|---|
| | | Symmetry | Range | Symmetry | Range | |
| $\mathcal{C}_{1e}^{-1}$ | $C_{1e}^{-1} = \frac{1}{2N} C_{1e}$ | WSWS | $0 \rightarrow N$ | WSWS | $0 \rightarrow N$ | $\mathcal{G}_{0,0}^{-1} \mathcal{E}_{WSWS}$ |
| $\mathcal{S}_{1e}^{-1}$ | $S_{1e}^{-1} = \frac{1}{2N} S_{1e}$ | WAWA | $1 \rightarrow N\text{-}1$ | WAWA | $1 \rightarrow N\text{-}1$ | $-j\mathcal{G}_{0,0}^{-1} \mathcal{E}_{WAWA}$ |
| $\mathcal{C}_{2e}^{-1}$ | $C_{2e}^{-1} = \frac{1}{2N} C_{3e}$ | WSWA | $0 \rightarrow N\text{-}1$ | HSHS | $0 \rightarrow N\text{-}1$ | $\mathcal{G}_{0,\frac{1}{2}}^{-1} \mathcal{E}_{WSWA}$ |
| $\mathcal{S}_{2e}^{-1}$ | $S_{2e}^{-1} = \frac{1}{2N} S_{3e}$ | WAWS | $1 \rightarrow N$ | HAHA | $0 \rightarrow N\text{-}1$ | $-j\mathcal{G}_{0,\frac{1}{2}}^{-1} \mathcal{E}_{WAWS}$ |
| $\mathcal{C}_{3e}^{-1}$ | $C_{3e}^{-1} = \frac{1}{2N} C_{2e}$ | HSHS | $0 \rightarrow N\text{-}1$ | WSWA | $0 \rightarrow N\text{-}1$ | $\mathcal{G}_{\frac{1}{2},0}^{-1} \mathcal{E}_{HSHS}$ |
| $\mathcal{S}_{3e}^{-1}$ | $S_{3e}^{-1} = \frac{1}{2N} S_{2e}$ | HAHA | $0 \rightarrow N\text{-}1$ | WAWS | $1 \rightarrow N$ | $-j\mathcal{G}_{\frac{1}{2},0}^{-1} \mathcal{E}_{HAHA}$ |
| $\mathcal{C}_{4e}^{-1}$ | $C_{4e}^{-1} = \frac{1}{2N} C_{4e}$ | HSHA | $0 \rightarrow N\text{-}1$ | HSHA | $0 \rightarrow N\text{-}1$ | $\mathcal{G}_{\frac{1}{2},\frac{1}{2}}^{-1} \mathcal{E}_{HSHA}$ |
| $\mathcal{S}_{4e}^{-1}$ | $S_{4e}^{-1} = \frac{1}{2N} S_{4e}$ | HAHS | $0 \rightarrow N\text{-}1$ | HAHS | $0 \rightarrow N\text{-}1$ | $-j\mathcal{G}_{\frac{1}{2},\frac{1}{2}}^{-1} \mathcal{E}_{HAHS}$ |
| $\mathcal{C}_{1o}^{-1}$ | $C_{1o}^{-1} = \frac{1}{2N-1} C_{1o}$ | WSHS | $0 \rightarrow N\text{-}1$ | WSHS | $0 \rightarrow N\text{-}1$ | $\mathcal{G}_{0,0}^{-1} \mathcal{E}_{WSHS}$ |
| $\mathcal{S}_{1o}^{-1}$ | $S_{1o}^{-1} = \frac{1}{2N-1} S_{1o}$ | WAHA | $1 \rightarrow N\text{-}1$ | WAHA | $1 \rightarrow N\text{-}1$ | $-j\mathcal{G}_{0,0}^{-1} \mathcal{E}_{WAHA}$ |
| $\mathcal{C}_{2o}^{-1}$ | $C_{2o}^{-1} = \frac{1}{2N-1} C_{3o}$ | WSHA | $0 \rightarrow N\text{-}1$ | HSWS | $0 \rightarrow N\text{-}1$ | $\mathcal{G}_{0,\frac{1}{2}}^{-1} \mathcal{E}_{WSHA}$ |
| $\mathcal{S}_{2o}^{-1}$ | $S_{2o}^{-1} = \frac{1}{2N-1} S_{3o}$ | WAHS | $1 \rightarrow N\text{-}1$ | HAWA | $0 \rightarrow N\text{-}2$ | $-j\mathcal{G}_{0,\frac{1}{2}}^{-1} \mathcal{E}_{WAHS}$ |
| $\mathcal{C}_{3o}^{-1}$ | $C_{3o}^{-1} = \frac{1}{2N-1} C_{2o}$ | HSWS | $0 \rightarrow N\text{-}1$ | WSHA | $0 \rightarrow N\text{-}1$ | $\mathcal{G}_{\frac{1}{2},0}^{-1} \mathcal{E}_{HSWS}$ |
| $\mathcal{S}_{3o}^{-1}$ | $S_{3o}^{-1} = \frac{1}{2N-1} S_{2o}$ | HAWA | $0 \rightarrow N\text{-}2$ | WAHS | $1 \rightarrow N\text{-}1$ | $-j\mathcal{G}_{\frac{1}{2},0}^{-1} \mathcal{E}_{HAWA}$ |
| $\mathcal{C}_{4o}^{-1}$ | $C_{4o}^{-1} = \frac{1}{2N-1} C_{4o}$ | HSWA | $0 \rightarrow N\text{-}2$ | HSWA | $0 \rightarrow N\text{-}2$ | $\mathcal{G}_{\frac{1}{2},\frac{1}{2}}^{-1} \mathcal{E}_{HSWA}$ |
| $\mathcal{S}_{4o}^{-1}$ | $S_{4o}^{-1} = \frac{1}{2N-1} S_{4o}$ | HAWS | $0 \rightarrow N\text{-}1$ | HAWS | $0 \rightarrow N\text{-}1$ | $-j\mathcal{G}_{\frac{1}{2},\frac{1}{2}}^{-1} \mathcal{E}_{HAWS}$ |

The formula for circular convolution, denoted by ©, is well-known [37]. Skew-circular convolution is expressed as

$$\begin{aligned} d(n) &= \theta(n) \circledS\, h(n) \\ &= \sum_{m=0}^{n} \theta(m) h(n-m) - \sum_{m=n+1}^{N-1} \theta(m) h(n-m+N). \end{aligned} \tag{32}$$

Martucci's key result [34] is that Eq. (31) is exactly equivalent to

$$d(n-n_0) = \mathcal{T}_3^{-1}\left\{\mathcal{T}_1\{\theta(n)\} \cdot \mathcal{T}_2\{h(n)\}\right\}, \tag{33}$$

where $\mathcal{T}_m\{\bullet\}$ denotes one of the 16 DTT operators. The value of $n_0$ equals either 0 or 1 indicating that in some cases the multiplication and inverse trigonometric transform operations introduce a one-sample delay in the output. Table 3 lists all forty cases from [34] of the symmetric convolution-multiplication property, and identifies those cases where the output is delayed by one sample.

Table 3. Forty Cases of Symmetric Convolution-Multiplication

| $N$ even, $M = 2N$ | | | | | | | $N$ odd, $M = 2N$-1 | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $\mathcal{E}_1$ | $\mathcal{E}_2$ | ⊛ | $\mathcal{T}_1$ | $\mathcal{T}_2$ | $\mathcal{T}_3^{-1}$ | $n_0$ | $\mathcal{E}_1$ | $\mathcal{E}_2$ | ⊛ | $\mathcal{T}_1$ | $\mathcal{T}_2$ | $\mathcal{T}_3^{-1}$ | $n_0$ |
| $\mathcal{E}_{WSWS}$ | $\mathcal{E}_{WSWS}$ | Ⓒ | $\mathcal{C}_{1e}$ | $\mathcal{C}_{1e}$ | $\mathcal{C}_{1e}^{-1}$ | 0 | $\mathcal{E}_{WSHS}$ | $\mathcal{E}_{WSHS}$ | Ⓒ | $\mathcal{C}_{1o}$ | $\mathcal{C}_{1o}$ | $\mathcal{C}_{1o}^{-1}$ | 0 |
| $\mathcal{E}_{WSWS}$ | $\mathcal{E}_{WAWA}$ | Ⓒ | $\mathcal{C}_{1e}$ | $\mathcal{S}_{1e}$ | $\mathcal{S}_{1e}^{-1}$ | 0 | $\mathcal{E}_{WSHS}$ | $\mathcal{E}_{WAHA}$ | Ⓒ | $\mathcal{C}_{1o}$ | $\mathcal{S}_{1o}$ | $\mathcal{S}_{1o}^{-1}$ | 0 |
| $\mathcal{E}_{WAWA}$ | $\mathcal{E}_{WAWA}$ | Ⓒ | $\mathcal{S}_{1e}$ | $\mathcal{S}_{1e}$ | $-\mathcal{C}_{1e}^{-1}$ | 0 | $\mathcal{E}_{WAHA}$ | $\mathcal{E}_{WAHA}$ | Ⓒ | $\mathcal{S}_{1o}$ | $\mathcal{S}_{1o}$ | $-\mathcal{C}_{1o}^{-1}$ | 0 |
| $\mathcal{E}_{HSHS}$ | $\mathcal{E}_{WSWS}$ | Ⓒ | $\mathcal{C}_{2e}$ | $\mathcal{C}_{1e}$ | $\mathcal{C}_{2e}^{-1}$ | 0 | $\mathcal{E}_{HSWS}$ | $\mathcal{E}_{WSHS}$ | Ⓒ | $\mathcal{C}_{2o}$ | $\mathcal{C}_{1o}$ | $\mathcal{C}_{2o}^{-1}$ | 0 |
| $\mathcal{E}_{HSHS}$ | $\mathcal{E}_{WAWA}$ | Ⓒ | $\mathcal{C}_{2e}$ | $\mathcal{S}_{1e}$ | $\mathcal{S}_{2e}^{-1}$ | 0 | $\mathcal{E}_{HSWS}$ | $\mathcal{E}_{WAHA}$ | Ⓒ | $\mathcal{C}_{2o}$ | $\mathcal{S}_{1o}$ | $\mathcal{S}_{2o}^{-1}$ | 0 |
| $\mathcal{E}_{HAHA}$ | $\mathcal{E}_{WSWS}$ | Ⓒ | $\mathcal{S}_{2e}$ | $\mathcal{C}_{1e}$ | $\mathcal{S}_{2e}^{-1}$ | 0 | $\mathcal{E}_{HAWA}$ | $\mathcal{E}_{WSHS}$ | Ⓒ | $\mathcal{S}_{2o}$ | $\mathcal{C}_{1o}$ | $\mathcal{S}_{2o}^{-1}$ | 0 |
| $\mathcal{E}_{HAHA}$ | $\mathcal{E}_{WAWA}$ | Ⓒ | $\mathcal{S}_{2e}$ | $\mathcal{S}_{1e}$ | $-\mathcal{C}_{2e}^{-1}$ | 0 | $\mathcal{E}_{HAWA}$ | $\mathcal{E}_{WAHA}$ | Ⓒ | $\mathcal{S}_{2o}$ | $\mathcal{S}_{1o}$ | $-\mathcal{C}_{2o}^{-1}$ | 0 |
| $\mathcal{E}_{HSHS}$ | $\mathcal{E}_{HSHS}$ | Ⓒ | $\mathcal{C}_{2e}$ | $\mathcal{C}_{2e}$ | $\mathcal{C}_{1e}^{-1}$ | 1 | $\mathcal{E}_{HSWS}$ | $\mathcal{E}_{HSWS}$ | Ⓒ | $\mathcal{C}_{2o}$ | $\mathcal{C}_{2o}$ | $\mathcal{C}_{1o}^{-1}$ | 1 |
| $\mathcal{E}_{HSHS}$ | $\mathcal{E}_{HAHA}$ | Ⓒ | $\mathcal{C}_{2e}$ | $\mathcal{S}_{2e}$ | $\mathcal{S}_{1e}^{-1}$ | 1 | $\mathcal{E}_{HSWS}$ | $\mathcal{E}_{HAWA}$ | Ⓒ | $\mathcal{C}_{2o}$ | $\mathcal{S}_{2o}$ | $\mathcal{S}_{1o}^{-1}$ | 1 |
| $\mathcal{E}_{HAHA}$ | $\mathcal{E}_{HAHA}$ | Ⓒ | $\mathcal{S}_{2e}$ | $\mathcal{S}_{2e}$ | $-\mathcal{C}_{1e}^{-1}$ | 1 | $\mathcal{E}_{HAWA}$ | $\mathcal{E}_{HAWA}$ | Ⓒ | $\mathcal{S}_{2o}$ | $\mathcal{S}_{2o}$ | $-\mathcal{C}_{1o}^{-1}$ | 1 |
| $\mathcal{E}_{WSWA}$ | $\mathcal{E}_{WSWA}$ | Ⓢ | $\mathcal{C}_{3e}$ | $\mathcal{C}_{3e}$ | $\mathcal{C}_{3e}^{-1}$ | 0 | $\mathcal{E}_{WSHA}$ | $\mathcal{E}_{WSHA}$ | Ⓢ | $\mathcal{C}_{3o}$ | $\mathcal{C}_{3o}$ | $\mathcal{C}_{3o}^{-1}$ | 0 |
| $\mathcal{E}_{WSWA}$ | $\mathcal{E}_{WAWS}$ | Ⓢ | $\mathcal{C}_{3e}$ | $\mathcal{S}_{3e}$ | $\mathcal{S}_{3e}^{-1}$ | 0 | $\mathcal{E}_{WSHA}$ | $\mathcal{E}_{WAHS}$ | Ⓢ | $\mathcal{C}_{3o}$ | $\mathcal{S}_{3o}$ | $\mathcal{S}_{3o}^{-1}$ | 0 |
| $\mathcal{E}_{WAWS}$ | $\mathcal{E}_{WAWS}$ | Ⓢ | $\mathcal{S}_{3e}$ | $\mathcal{S}_{3e}$ | $-\mathcal{C}_{3e}^{-1}$ | 0 | $\mathcal{E}_{WAWS}$ | $\mathcal{E}_{WAHS}$ | Ⓢ | $\mathcal{S}_{3o}$ | $\mathcal{S}_{3o}$ | $-\mathcal{C}_{3o}^{-1}$ | 0 |
| $\mathcal{E}_{HSHA}$ | $\mathcal{E}_{WSWA}$ | Ⓢ | $\mathcal{C}_{4e}$ | $\mathcal{C}_{3e}$ | $\mathcal{C}_{4e}^{-1}$ | 0 | $\mathcal{E}_{HSWA}$ | $\mathcal{E}_{WSHA}$ | Ⓢ | $\mathcal{C}_{4o}$ | $\mathcal{C}_{3o}$ | $\mathcal{C}_{4o}^{-1}$ | 0 |
| $\mathcal{E}_{HSHA}$ | $\mathcal{E}_{WAWS}$ | Ⓢ | $\mathcal{C}_{4e}$ | $\mathcal{S}_{3e}$ | $\mathcal{S}_{4e}^{-1}$ | 0 | $\mathcal{E}_{HSWA}$ | $\mathcal{E}_{WAHS}$ | Ⓢ | $\mathcal{C}_{4o}$ | $\mathcal{S}_{3o}$ | $\mathcal{S}_{4o}^{-1}$ | 0 |
| $\mathcal{E}_{HAHS}$ | $\mathcal{E}_{WSWA}$ | Ⓢ | $\mathcal{S}_{4e}$ | $\mathcal{C}_{3e}$ | $\mathcal{S}_{4e}^{-1}$ | 0 | $\mathcal{E}_{HAWS}$ | $\mathcal{E}_{WSHA}$ | Ⓢ | $\mathcal{S}_{4o}$ | $\mathcal{C}_{3o}$ | $\mathcal{S}_{4o}^{-1}$ | 0 |
| $\mathcal{E}_{HAHS}$ | $\mathcal{E}_{WAWS}$ | Ⓢ | $\mathcal{S}_{4e}$ | $\mathcal{S}_{3e}$ | $-\mathcal{C}_{4e}^{-1}$ | 0 | $\mathcal{E}_{HAWS}$ | $\mathcal{E}_{WAHS}$ | Ⓢ | $\mathcal{S}_{4o}$ | $\mathcal{S}_{3o}$ | $-\mathcal{C}_{4o}^{-1}$ | 0 |
| $\mathcal{E}_{HSHA}$ | $\mathcal{E}_{HSHA}$ | Ⓢ | $\mathcal{C}_{4e}$ | $\mathcal{C}_{4e}$ | $\mathcal{C}_{3e}^{-1}$ | 1 | $\mathcal{E}_{HSWA}$ | $\mathcal{E}_{HSWA}$ | Ⓢ | $\mathcal{C}_{4o}$ | $\mathcal{C}_{4o}$ | $\mathcal{C}_{3o}^{-1}$ | 1 |
| $\mathcal{E}_{HSHA}$ | $\mathcal{E}_{HAHS}$ | Ⓢ | $\mathcal{C}_{4e}$ | $\mathcal{S}_{4e}$ | $\mathcal{S}_{3e}^{-1}$ | 1 | $\mathcal{E}_{HSWA}$ | $\mathcal{E}_{HAWS}$ | Ⓢ | $\mathcal{C}_{4o}$ | $\mathcal{S}_{4o}$ | $\mathcal{S}_{3o}^{-1}$ | 1 |
| $\mathcal{E}_{HAHS}$ | $\mathcal{E}_{HAHS}$ | Ⓢ | $\mathcal{S}_{4e}$ | $\mathcal{S}_{4e}$ | $-\mathcal{C}_{3e}^{-1}$ | 1 | $\mathcal{E}_{HAWS}$ | $\mathcal{E}_{HAWS}$ | Ⓢ | $\mathcal{S}_{4o}$ | $\mathcal{S}_{4o}$ | $-\mathcal{C}_{3o}^{-1}$ | 1 |

Each group of transforms in an entry of Table 3 performs convolution in the sequence domain by transforming the two sequences, multiplying the results in the transform domain, and then taking the inverse transform to produce an answer in the sequence domain. Using the sym-

metric convolution-multiplication property results in a savings in computational complexity as fast algorithms exist for the DTTs [5], [39], [50]. The algorithms have the same computational complexity as the fast Fourier transform.

The symmetric convolution-multiplication property of the DTTs requires implicit symmetry in the sequences being convolved. The underlying symmetry is analogous to the implied periodicity required of sequences that are circularly convolved [37]. Martucci states [34] that the symmetric convolution-multiplication property he derives for one-dimensional sequences extends to asymmetric sequences if they are first decomposed into their symmetric and antisymmetric parts. This decomposition is straightforward in one dimension, but can be complicated in multiple dimensions because an asymmetric sequence must be decomposed into its symmetric and antisymmetric parts in each dimension across multiple lines, planes, or hyperplanes of symmetry [34].

The first section of this background chapter described how DFT matrices diagonalize circular convolution matrices. The next chapter shows how to derive similar relationships where the DTT matrices presented in this last section diagonalize symmetric convolution matrices. When represented by matrices, the symmetric convolution-multiplication property extends more easily to both multidimensional and asymmetric sequences. The new matrix form of the symmetric convolution-multiplication property then becomes the basis for deriving inverse and scalar Wiener filters in the trigonometric transform domain. These filters are shown to be similar but improved versions of the Fourier domain representations presented in the second section of this chapter.

# III. *Symmetric Convolution of Asymmetric Multidimensional Sequences*

An alternate method of deriving Martucci's [34] symmetric convolution-multiplication property of discrete trigonometric transforms is presented in this chapter. A vector-matrix form of the property is produced by this alternate method [9]. A new procedure is demonstrated of how the convolutional forms of trigonometric transforms can operate as matrices which diagonalize symmetric convolution matrices [11]. The diagonalized forms of the symmetric convolution matrices then allow the symmetric convolution-multiplication property to be easily extended to multidimensional and asymmetric sequences.

## 3.1 *Vector-Matrix Derivation of the Symmetric Convolution-Multiplication Property*

It is possible to derive the symmetric convolution-multiplication property for discrete trigonometric transforms (DTTs) using vector-matrix methods [9]. The result of the derivation is very similar to Hunt's result [23] for the discrete Fourier transform (DFT) presented as background in the previous chapter. This alternate derivation provides a new look at the symmetric convolution-multiplication property for the DTTs, just as Hunt's result provided a vector-matrix proof of the circular convolution-multiplication property for the DFT. The derivation which follows relies heavily on his key result expressed as Eq. (1).

The first step in the vector-matrix derivation of the symmetric convolution-multiplication property is to recognize the matrix relationship between the DFT and the generalized DFT (GDFT). An $M \times M$ DFT matrix, $\boldsymbol{W}_M^{-1}$, defined in Eq. (2) is expressible in terms of each of the four forward and inverse GDFT matrices, $\boldsymbol{G}_{a,b,M}$ and $\boldsymbol{G}_{a,b,M}^{-1}$, from Eq. (30) where $a$ and $b$ equal 0 or $\frac{1}{2}$. The DFT matrix relates to each of these GDFT matrices through an $M \times M$ diagonal

matrix $\boldsymbol{D}_M$ with elements $\left[\boldsymbol{D}_M\right]_{mm} = \exp\left\{j\frac{\pi}{M}m\right\}$ for $m = 0,\ 1,\ \ldots,\ M-1$. The sizes of the transform matrices and the matrix, $\boldsymbol{D}_M$, which relates them are all $M \times M$ since they operate on symmetric extensions of the sequences to be convolved. Recall that the original sequences to be convolved had length $N$, but the symmetric extension required to perform symmetric convolution increases their length to $M = 2N$ for $N$ even or $M = 2N-1$ for $N$ odd. The relationship among the four forward GDFTs is

$$\boldsymbol{W}_M^{-1} = \boldsymbol{G}_{0,0,M} = \boldsymbol{G}_{\frac{1}{2},0,M}\boldsymbol{D}_M = \boldsymbol{D}_M\boldsymbol{G}_{0,\frac{1}{2},M} = e^{j\frac{\pi}{2M}}\boldsymbol{D}_M\boldsymbol{G}_{\frac{1}{2},\frac{1}{2},M}\boldsymbol{D}_M . \tag{34}$$

The inverse DFT matrix is expressible in terms of the inverse GDFT matrix for each case by inverting the above expressions. All inverses exist so that

$$\boldsymbol{W}_M = \boldsymbol{G}_{0,0,M}^{-1} = \boldsymbol{D}_M^{-1}\boldsymbol{G}_{\frac{1}{2},0,M}^{-1} = \boldsymbol{G}_{0,\frac{1}{2},M}^{-1}\boldsymbol{D}_M^{-1} = e^{-j\frac{\pi}{2M}}\boldsymbol{D}_M^{-1}\boldsymbol{G}_{\frac{1}{2},\frac{1}{2},M}^{-1}\boldsymbol{D}_M^{-1} . \tag{35}$$

An additional convolution-multiplication property exists for the skew-circular convolution operation $\boldsymbol{d} = \boldsymbol{h} \circledS \theta = \boldsymbol{H}_S\theta$, where $\boldsymbol{H}_S$ is an $N \times N$ skew-circulant matrix. That is,

$$\boldsymbol{H}_S = \begin{bmatrix} h(0) & -h(N-1) & \cdots & -h(1) \\ h(1) & h(0) & \cdots & -h(2) \\ \vdots & \vdots & \ddots & \vdots \\ h(N-1) & h(N-2) & \cdots & h(0) \end{bmatrix}. \tag{36}$$

As mentioned previously, skew-circular convolution is the underlying form of convolution in half of the forty cases of symmetric convolution [34]. The convolution-multiplication property for skew-circular convolution is an extension of a result of Vernet's [49]. His result begins with the GDFT of $\theta(n)$ for $a = \frac{1}{2}$ and $b = 0$,

$$\begin{aligned} \vartheta_O(k) &= \mathcal{G}_{\frac{1}{2},0}\{\theta(n)\} \\ &= \sum_{n=0}^{N-1}\theta(n)\exp\left(-j2\pi\frac{(k+\frac{1}{2})n}{N}\right), \end{aligned} \tag{37}$$

for $k = 0, 1, \ldots, N-1$. The subscript '$O$' in the term $\mathcal{G}_O(k)$ indicates the transform domain of the *odd DFT* as Vernet defines this transform. The skew-circulant convolution of two sequences, $h(n)$ and $\theta(n)$, is exactly equivalent to the inverse odd DFT of the product of their transforms, $\mathcal{H}_O(k)$ and $\mathcal{G}_O(k)$, or equivalently,

$$\begin{aligned} d(n) &= \mathcal{G}_{\frac{1}{2},0}^{-1}\left\{\mathcal{H}_O(k)\mathcal{G}_O(k)\right\} \\ &= \frac{1}{N}\sum_{k=0}^{N-1}\mathcal{H}_O(k)\mathcal{G}_O(k)\exp\left(j2\pi\frac{(k+\frac{1}{2})n}{N}\right), \end{aligned} \tag{38}$$

for $n = 0, 1, \ldots, N-1$. In terms of matrices, Martucci [34] generalizes Vernet's result for the inverse odd DFT to $\boldsymbol{G}_{\frac{1}{2},0,N}^{-1} = \frac{1}{N}\boldsymbol{G}_{\frac{1}{2},0,N}^{H} = \frac{1}{N}\boldsymbol{G}_{0,\frac{1}{2},N}^{*}$.

Equation (38) implies that an odd DFT matrix diagonalizes a skew-circulant matrix $\boldsymbol{H}_S$ through the relation $\boldsymbol{H}_S = \boldsymbol{V}_N\mathcal{H}_O\boldsymbol{V}_N^{-1}$. The odd DFT matrix , $\boldsymbol{V}_N^{-1}$, equals the $N \times N$ GDFT matrix, $\boldsymbol{G}_{\frac{1}{2},0,N}$, where $a = \frac{1}{2}$ and $b = 0$. The matrix $\mathcal{H}_O$ is diagonal with the elements of $\boldsymbol{G}_{\frac{1}{2},0,N}\boldsymbol{h}$ along the diagonal. Thus, for skew-circular convolution, $\boldsymbol{G}_{\frac{1}{2},0,N}\boldsymbol{d} = \left(\boldsymbol{G}_{\frac{1}{2},0,N}\boldsymbol{h}\right) \odot \left(\boldsymbol{G}_{\frac{1}{2},0,N}\theta\right)$.

The matrix $\boldsymbol{V}_M^{-1}$ is the odd DFT matrix which operates on symmetrically extended sequences of length $M$. This matrix will diagonalize an $M \times M$ skew-circulant convolution matrix. Like the DFT matrix in Eqs. (34) and (35), the matrix $\boldsymbol{V}_M^{-1}$ is expressible in terms of each of the four special cases of the $M \times M$ GDFT for which $a$ and $b$ equal 0 or $\frac{1}{2}$. Here the relationships are

$$\boldsymbol{V}_M^{-1} = \boldsymbol{G}_{0,0,M}\boldsymbol{D}_M^{-1} = \boldsymbol{G}_{\frac{1}{2},0,M} = \boldsymbol{D}_M\boldsymbol{G}_{0,\frac{1}{2},M}\boldsymbol{D}_M^{-1} = e^{j\frac{\pi}{2M}}\boldsymbol{D}_M\boldsymbol{G}_{\frac{1}{2},\frac{1}{2},M}. \tag{39}$$

Again all inverses exist, so that

$$\boldsymbol{V}_M = \boldsymbol{D}_M\boldsymbol{G}_{0,0,M}^{-1} = \boldsymbol{G}_{\frac{1}{2},0,M}^{-1} = \boldsymbol{D}_M\boldsymbol{G}_{0,\frac{1}{2},M}^{-1}\boldsymbol{D}_M^{-1} = e^{-j\frac{\pi}{2M}}\boldsymbol{G}_{\frac{1}{2},\frac{1}{2},M}^{-1}\boldsymbol{D}_M^{-1}. \tag{40}$$

The matrices in Eqs. (34) and (39) and their inverses in Eqs. (35) and (40) will pre- and post-multiply the circulant and skew-circulant matrices underlying symmetric convolution as part of deriving the symmetric convolution-multiplication property of the DTTs. The result of this multiplication must be diagonal for each of the forty cases of symmetric convolution to show that the multiplication property holds. First, however, symmetric convolution must be expressed as a matrix operation. Equation (31) which used operator notation is equivalent to

$$\boldsymbol{d} = \boldsymbol{R}_L \boldsymbol{H}_M \boldsymbol{E}_1 \theta, \tag{41}$$

which is expressed using vector-matrix notation. The vectors $\theta$ and $\boldsymbol{d}$ are $L \times 1$ input and output vectors. The matrix $\boldsymbol{E}_1$ is an $M \times L$ symmetric extension matrix. The matrix $\boldsymbol{H}_M$ is an $M \times M$ circulant or skew-circulant matrix with the vector $\boldsymbol{E}_2 \boldsymbol{h}$ as its first column. The matrix $\boldsymbol{R}_L = \left[\boldsymbol{I}_L \;\; \boldsymbol{0}\right]$, where $\boldsymbol{I}_L$ is an $L \times L$ identity matrix, and $\boldsymbol{0}$ is an appropriately-sized zero matrix to make $\boldsymbol{R}_L$ an $L \times M$ matrix. The length $L$ equals $N-1$, $N$, or $N+1$ depending on the type of symmetry implied and the different sizes for input and output sequences listed in Table 1. As before, the length $M$ equals $2N$ for $N$ even or $2N-1$ for $N$ odd.

All of the symmetric extensions, shifts, additions, and multiplications of symmetric convolution in Eq. (41) can be captured in a single $L \times L$ matrix, $\boldsymbol{H}_{SC} = \boldsymbol{R}_L \boldsymbol{H}_M \boldsymbol{E}_1$, so that $\boldsymbol{d} = \boldsymbol{H}_{SC} \theta$. The subscript *'SC'* denotes symmetric convolution, which is generalized in this derivation to any one of the forty cases of symmetric convolution. The form of the matrix $\boldsymbol{H}_{SC}$ will be different for each of the forty different cases. The goal of this alternate method of deriving the symmetric convolution-multiplication property is to show that an $L \times L$ diagonal matrix form, $\mathcal{H}_T$, exists for the symmetric convolution matrix, $\boldsymbol{H}_{SC}$. The subscript *'T'* denotes that the diagonal matrix $\mathcal{H}_T$ exists in the trigonometric transform domain. It will be shown that $L \times L$ trigonometric

transform matrices, represented, in general, by the matrix $\boldsymbol{T}_L$, cause this diagonalization. The matrix $\boldsymbol{T}_L$ can represent any one of the sixteen DTTs.

Before proceeding with the derivation, it is useful to present a summary of the three types of one-dimensional transforms discussed thus far in tabular form. This summary for the DFT, the odd DFT, and an arbitrary DTT appears in Table 4. All of the matrices related to the DFT

Table 4. One-Dimensional Transforms

| Type | Matrix Representation | Underlying Form of Convolution | Convolution Matrix | Diagonal Form in Transform Domain |
|---|---|---|---|---|
| DFT | $\boldsymbol{W}_M^{-1}$ | circular | $\boldsymbol{H}_C$ | $\mathscr{H}_F$ |
| odd-DFT | $\boldsymbol{V}_M^{-1}$ | skew-circular | $\boldsymbol{H}_S$ | $\mathscr{H}_O$ |
| DTT | $\boldsymbol{T}_L$ | symmetric | $\boldsymbol{H}_{SC}$ | $\mathscr{H}_T$ |

and the odd DFT listed in Table 4 have dimension $M \times M$ because they operate within the definition of symmetric convolution on symmetric extensions of the sequences being convolved. The trigonometric transform matrix, $\boldsymbol{T}_L$, and the matrices $\boldsymbol{H}_{SC}$ and $\mathscr{H}_T$ all have dimension $L \times L$ because the individual DTT matrices can have dimension $N-1 \times N-1$, $N \times N$, or $N+1 \times N+1$ depending on the choice of transform from Table 1 in the previous chapter. The dimension of each of the sixteen transforms is based on the underlying type of symmetry present in the sequence being transformed. Table 1 also listed the underlying symmetry and sequence length for each of the sixteen DTTs.

To proceed with the vector-matrix derivation of the symmetric convolution-multiplication property, the diagonal form of the matrix $\boldsymbol{H}_M$ must next be substituted into Eq. (41). The matrix $\boldsymbol{H}_M$ will be either circulant or skew-circulant depending on the underlying form of convolution in the particular case of symmetric convolution from Table 3 that Eq. (41) represents. Recall that Eq. (41) generically represents any one of the forty cases. Thus the appropriate substitution for

$\boldsymbol{H}_M$ in Eq. (41) will be $\boldsymbol{W}_M \mathcal{H}_F \boldsymbol{W}_M^{-1}$, where $\mathcal{H}_F = \text{diag}\left\{\boldsymbol{W}_M^{-1}\boldsymbol{E}_2\boldsymbol{h}\right\}$ if the underlying form of convolution is circular. The substitution for $\boldsymbol{H}_M$ in Eq. (41) will be $\boldsymbol{V}_M \mathcal{H}_O \boldsymbol{V}_M^{-1}$, where $\mathcal{H}_O = \text{diag}\left\{\boldsymbol{V}_M^{-1}\boldsymbol{E}_2\boldsymbol{h}\right\}$ if the underlying form of convolution is skew-circular. The $\text{diag}\{\bullet\}$ operation creates a diagonal matrix from a vector. Any of the equivalent expressions from Eqs. (34), (35), (39), or (40) then become allowable substitutions for $\boldsymbol{W}_M^{-1}$, $\boldsymbol{W}_M$, $\boldsymbol{V}_M^{-1}$, or $\boldsymbol{V}_M$ in the result. With these substitutions, Eq. (41) becomes

$$\begin{aligned} \boldsymbol{d} &= \boldsymbol{R}_L \boldsymbol{G}_3^{-1} \mathcal{H}_M \boldsymbol{G}_1 \boldsymbol{E}_1 \theta \\ &= \boldsymbol{R}_L \boldsymbol{G}_3^{-1} \left[\text{diag}\left\{\boldsymbol{G}_2 \boldsymbol{E}_2 \boldsymbol{h}\right\} \boldsymbol{G}_1 \boldsymbol{E}_1 \theta\right] \\ &= \boldsymbol{R}_L \boldsymbol{G}_3^{-1} \left[\boldsymbol{G}_2 \boldsymbol{E}_2 \boldsymbol{h} \odot \boldsymbol{G}_1 \boldsymbol{E}_1 \theta\right]. \end{aligned} \tag{42}$$

The matrix $\mathcal{H}_M$ in Eq. (42) is either $\mathcal{H}_F$ or $\mathcal{H}_O$ depending on the underlying form of convolution. The three matrices, $\boldsymbol{G}_m$, each represent one of the four special cases of an $M \times M$ GDFT matrix for which $a$ and $b$ equal 0 or $\frac{1}{2}$. In some cases the matrix $\boldsymbol{G}_m$ may include multiplication by a factor of $j$ or $-j$. This factor is necessary because of the relationship between each trigonometric transform and the GDFT.

The relationship between each DTT operator, $\mathcal{T}\{\bullet\}$, and the GDFT operator, $\mathcal{G}_{a,b}\{\bullet\}$, is shown in Tables 1 and 2 for the forward and inverse transforms. One of Martucci's results [34] is that the DTT of a sequence can be expressed as a GDFT operating on a symmetrically extended version of the sequence. That is, $\mathcal{T}\{x(n)\} = \mathcal{G}_{a,b}\{\varepsilon_{SE}\{x(n)\}\}$, where $\varepsilon_{SE}\{\bullet\}$ is one of the sixteen symmetric extension operators, hence bearing the subscript *'SE'*. The example was given in the previous chapter that $\mathcal{C}_{1e}\{x(n)\} = \mathcal{G}_{0,0}\{\varepsilon_{WSWS}\{x(n)\}\}$. Another example which shows the

equivalence of a DTT to the GDFT is $\mathcal{S}_{4o}^{-1}\{x(n)\} = -j\mathcal{G}_{\frac{1}{2},\frac{1}{2}}^{-1}\{\varepsilon_{HAWS}\{x(n)\}\}$ from Table 2. Here the need for multiplication by a factor of $-j$ is illustrated.

An equivalence between DTT and GDFT matrices exists that is similar to the equivalence between DTT and GDFT operators. If the sequence $x(n)$ is expressed as an $L \times 1$ vector $\boldsymbol{x}$, then it can be symmetrically extended by the $M \times L$ matrix $\boldsymbol{E}_{SE}$ as demonstrated in the previous chapter. This extension produces the $M \times 1$ vector $\tilde{\boldsymbol{x}} = \boldsymbol{E}_{SE}\boldsymbol{x}$. Now multiplying the vector $\tilde{\boldsymbol{x}}$ by an $M \times M$ GDFT matrix, $\boldsymbol{G}_{a,b,M}$, will be exactly equivalent to multiplying the vector $\boldsymbol{x}$ by the $M \times L$ extended DTT matrix $\tilde{\boldsymbol{T}}$. That is, $\tilde{\boldsymbol{T}}\boldsymbol{x} = \boldsymbol{G}_{a,b,M}\tilde{\boldsymbol{x}} = \boldsymbol{G}_{a,b,M}\boldsymbol{E}_{SE}\boldsymbol{x}$, so that the vector-matrix equivalent form of $\mathcal{T}\{x(n)\} = \mathcal{G}_{a,b}\{\varepsilon_{SE}\{x(n)\}\}$ is $\tilde{\boldsymbol{T}}\boldsymbol{x} = \boldsymbol{G}_{a,b,M}\boldsymbol{E}_{SE}\boldsymbol{x}$. The tilde, ~, on the transform matrix, $\tilde{\boldsymbol{T}}$, indicates that it is an extended-length transform. Normally a DTT matrix, $\boldsymbol{T}_L$, has dimension $L \times L$, but its $M \times L$ extended version, $\tilde{\boldsymbol{T}}$, has twice as many rows. The equivalence between DTT and GDFT matrices allows substitutions of the form $\tilde{\boldsymbol{T}}_m = \boldsymbol{G}_m\boldsymbol{E}_m$, to be made in Eq. (42). These substitutions produce

$$\boldsymbol{d} = \boldsymbol{R}_L\boldsymbol{G}_3^{-1}\left[\tilde{\boldsymbol{T}}_2\boldsymbol{h} \odot \tilde{\boldsymbol{T}}_1\theta\right]. \tag{43}$$

The portion of Eq. (43) inside the brackets results in an $M \times 1$ vector because of the point-wise multiplication. The vectors $\theta$ and $\boldsymbol{h}$ both have dimension $L \times 1$, but are transformed into $M \times 1$ vectors by the $M \times L$ transformation matrices $\tilde{\boldsymbol{T}}_1$ and $\tilde{\boldsymbol{T}}_2$. The $M \times 1$ vectors $\tilde{\boldsymbol{T}}_1\theta$ and $\tilde{\boldsymbol{T}}_2\boldsymbol{h}$ will both represent symmetric periodic sequences (SPS's). They will be SPS's because the DTTs require an implied symmetry associated with both their input and output sequences. Just like the input and output sequences to a DFT are periodic, the input and output sequences to a DTT are symmetric periodic. The product of any two SPS's is itself an SPS, so that the point-wise product

$\tilde{\boldsymbol{T}}_2\boldsymbol{h} \odot \tilde{\boldsymbol{T}}_1\theta$ will also represent an SPS. Because the product $\tilde{\boldsymbol{T}}_2\boldsymbol{h} \odot \tilde{\boldsymbol{T}}_1\theta$ has dimension $M \times 1$, it will have twice as many samples as the $L$ samples which are needed to completely characterize it as an SPS. The $L$ samples which completely characterize the vector $\tilde{\boldsymbol{T}}_2\boldsymbol{h} \odot \tilde{\boldsymbol{T}}_1\theta$ are analogous to the samples which characterize a periodic sequence by its fundamental period. Since the vector $\tilde{\boldsymbol{T}}_2\boldsymbol{h} \odot \tilde{\boldsymbol{T}}_1\theta$ can be completely characterized by $L$ samples, the $M \times L$ matrices $\tilde{\boldsymbol{T}}_1$ and $\tilde{\boldsymbol{T}}_2$ in Eq. (43) can be reduced to their standard $L \times L$ size represented by $\boldsymbol{T}_1$ and $\boldsymbol{T}_2$. A third $M \times L$ symmetric extension matrix, $\boldsymbol{E}_3$, can then symmetrically extend the result of the $L \times 1$ point-wise multiplication $\boldsymbol{T}_2\boldsymbol{h} \odot \boldsymbol{T}_1\theta$. Making the substitution $\tilde{\boldsymbol{T}}_2\boldsymbol{h} \odot \tilde{\boldsymbol{T}}_1\theta = \boldsymbol{E}_3[\boldsymbol{T}_2\boldsymbol{h} \odot \boldsymbol{T}_1\theta]$ in Eq. (43) produces

$$\begin{aligned} \boldsymbol{d} &= \boldsymbol{R}_L\boldsymbol{G}_3^{-1}\boldsymbol{E}_3[\boldsymbol{T}_2\boldsymbol{h} \odot \boldsymbol{T}_1\theta] \\ &= \boldsymbol{R}_L\tilde{\boldsymbol{T}}_3^{-1}[\boldsymbol{T}_2\boldsymbol{h} \odot \boldsymbol{T}_1\theta], \end{aligned} \tag{44}$$

by recognizing that $\boldsymbol{G}_3^{-1}\boldsymbol{E}_3 = \tilde{\boldsymbol{T}}_3^{-1}$. The effect of the $L \times M$ matrix $\boldsymbol{R}_L$ is to retain the first $L$ samples of the $M \times 1$ vector $\tilde{\boldsymbol{T}}_3^{-1}[\boldsymbol{T}_2\boldsymbol{h} \odot \boldsymbol{T}_1\theta]$. These $L$ samples are the same which result from reducing the $M \times L$ matrix $\tilde{\boldsymbol{T}}_3$ in Eq. (44) to its standard $L \times L$ size represented by $\boldsymbol{T}_3$. Thus

$$\boldsymbol{d} = \boldsymbol{T}_3^{-1}[\boldsymbol{T}_2\boldsymbol{h} \odot \boldsymbol{T}_1\theta], \tag{45}$$

where $\boldsymbol{T}_1$, $\boldsymbol{T}_2$, and $\boldsymbol{T}_3^{-1}$ represent $L \times L$ trigonometric forward and inverse transforms. Equation (45) is the symmetric convolution-multiplication property of the discrete trigonometric transforms expressed in vector-matrix notation. An alternate and sometimes more useful method of expressing the result in Eq. (45) is

$$\boldsymbol{d} = \boldsymbol{T}_3^{-1}\mathscr{H}_T\boldsymbol{T}_1\theta. \tag{46}$$

The $L \times L$ diagonal matrix $\mathcal{H}_T$ equals $\text{diag}\{\boldsymbol{T}_2\boldsymbol{h}\}$, where the subscript '$T$' again indicates the trigonometric transform domain.

The preceding general procedure allows for the derivation of any one of the forty versions of symmetric convolution in Table 3 using vector notation. The key to deriving any particular case is to make the appropriate substitutions for each $\boldsymbol{E}_m$ and $\boldsymbol{G}_m$ in Eqs. (41) - (44) which will yield the correct three transforms, $\boldsymbol{T}_m$, in Eqs. (45) and (46). The appropriate substitutions for $\boldsymbol{E}_m$ and $\boldsymbol{G}_m$ for each of the forty cases appear in Table 5. Note that the entries in the table temporarily suppress the subscript '$M$' indicating the dimension of the transform since all GDFT entries have dimension $M \times M$. Note that some inverse transforms in the table have the form $\boldsymbol{G}_{0,1,M}^{-1}$ or $\boldsymbol{G}_{\frac{1}{2},1,M}^{-1}$, where the '1' indicates a whole-sample delay. These delaying forms of the GDFT result from the expressions $\boldsymbol{G}_{0,0,M}^{-1}\boldsymbol{D}_M^2 = \boldsymbol{G}_{0,1,M}^{-1}$ and $\boldsymbol{G}_{\frac{1}{2},0,M}^{-1}e^{j\frac{\pi}{M}}\boldsymbol{D}_M^2 = \boldsymbol{G}_{\frac{1}{2},1,M}^{-1}$. They occur in each case where a one-sample delay occurs in the symmetric convolution output in Eq. (33) for the cases where $n_0 = 1$.

As an example of how to derive one of the forty types of symmetric convolution, consider the case where the $N \times 1$ vectors $\boldsymbol{h}$ and $\theta$ have WSWS and HSHS symmetry, respectively. In this case

$$\boldsymbol{d} = \boldsymbol{R}_L\boldsymbol{H}_C\boldsymbol{E}_{HSHS}\theta, \tag{47}$$

where $\boldsymbol{H}_C$ is an $M \times M$ circulant matrix with the $M \times 1$ vector $\boldsymbol{E}_{WSWS}\boldsymbol{h}$ as its first column. The matrix $\boldsymbol{H}_C$ is then replaced by $\boldsymbol{W}_M\mathcal{H}_F\boldsymbol{W}_M^{-1}$, where $\mathcal{H}_F = \text{diag}\{\boldsymbol{W}_M^{-1}\boldsymbol{E}_{WSWS}\boldsymbol{h}\}$. The facts from Eqs. (34) and (35) that $\boldsymbol{W}_M = \boldsymbol{G}_{0,\frac{1}{2},M}^{-1}\boldsymbol{D}_M^{-1}$, $\boldsymbol{W}_M^{-1} = \boldsymbol{D}_M\boldsymbol{G}_{0,\frac{1}{2},M}$, and $\boldsymbol{W}_M^{-1} = \boldsymbol{G}_{0,0,M}$ allow $\boldsymbol{H}_C$ to be

Table 5. Substitutions in Eqs. (41) - (44) used to Derive the Forty Cases of Symmetric Convolution

| $N$ even, $M = 2N$ | | | | | | $N$ odd, $M = 2N$-1 | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $E_1$ | $E_2$ | $E_3$ | $G_1$ | $G_2$ | $G_3^{-1}$ | $E_1$ | $E_2$ | $E_3$ | $G_1$ | $G_2$ | $G_3^{-1}$ |
| $E_{WSWS}$ | $E_{WSWS}$ | $E_{WSWS}$ | $G_{0,0}$ | $G_{0,0}$ | $G_{0,0}^{-1}$ | $E_{WSHS}$ | $E_{WSHS}$ | $E_{WSHS}$ | $G_{0,0}$ | $G_{0,0}$ | $G_{0,0}^{-1}$ |
| $E_{WSWS}$ | $E_{WAWA}$ | $E_{WAWA}$ | $G_{0,0}$ | $jG_{0,0}$ | $-jG_{0,0}^{-1}$ | $E_{WSHS}$ | $E_{WAHA}$ | $E_{WAHA}$ | $G_{0,0}$ | $jG_{0,0}$ | $-jG_{0,0}^{-1}$ |
| $E_{WAWA}$ | $E_{WAWA}$ | $E_{WSWS}$ | $jG_{0,0}$ | $jG_{0,0}$ | $-G_{0,0}^{-1}$ | $E_{WAHA}$ | $E_{WAHA}$ | $E_{WSHS}$ | $jG_{0,0}$ | $jG_{0,0}$ | $-G_{0,0}^{-1}$ |
| $E_{HSHS}$ | $E_{WSWS}$ | $E_{WSWA}$ | $G_{0,\frac{1}{2}}$ | $G_{0,0}$ | $G_{0,\frac{1}{2}}^{-1}$ | $E_{HSWS}$ | $E_{WSHS}$ | $E_{WSHA}$ | $G_{0,\frac{1}{2}}$ | $G_{0,0}$ | $G_{0,\frac{1}{2}}^{-1}$ |
| $E_{HSHS}$ | $E_{WAWA}$ | $E_{WAWS}$ | $G_{0,\frac{1}{2}}$ | $jG_{0,0}$ | $-jG_{0,\frac{1}{2}}^{-1}$ | $E_{HSWS}$ | $E_{WAHA}$ | $E_{WAHS}$ | $G_{0,\frac{1}{2}}$ | $jG_{0,0}$ | $-jG_{0,\frac{1}{2}}^{-1}$ |
| $E_{HAHA}$ | $E_{WSWS}$ | $E_{WAWS}$ | $jG_{0,\frac{1}{2}}$ | $G_{0,0}$ | $-jG_{0,\frac{1}{2}}^{-1}$ | $E_{HAWA}$ | $E_{WSHS}$ | $E_{WAHS}$ | $jG_{0,\frac{1}{2}}$ | $G_{0,0}$ | $-jG_{0,\frac{1}{2}}^{-1}$ |
| $E_{HAHA}$ | $E_{WAWA}$ | $E_{WSWA}$ | $jG_{0,\frac{1}{2}}$ | $jG_{0,0}$ | $-G_{0,\frac{1}{2}}^{-1}$ | $E_{HAWA}$ | $E_{WAHA}$ | $E_{WSHA}$ | $jG_{0,\frac{1}{2}}$ | $jG_{0,0}$ | $-G_{0,\frac{1}{2}}^{-1}$ |
| $E_{HSHS}$ | $E_{HSHS}$ | $E_{WSWS}$ | $G_{0,\frac{1}{2}}$ | $G_{0,\frac{1}{2}}$ | $G_{0,1}^{-1}$ | $E_{HSWS}$ | $E_{HSWS}$ | $E_{WSHS}$ | $G_{0,\frac{1}{2}}$ | $G_{0,\frac{1}{2}}$ | $G_{0,1}^{-1}$ |
| $E_{HSHS}$ | $E_{HAHA}$ | $E_{WAWA}$ | $G_{0,\frac{1}{2}}$ | $jG_{0,\frac{1}{2}}$ | $-jG_{0,1}^{-1}$ | $E_{HSWS}$ | $E_{HAWA}$ | $E_{WAHA}$ | $G_{0,\frac{1}{2}}$ | $jG_{0,\frac{1}{2}}$ | $-jG_{0,1}^{-1}$ |
| $E_{HAHA}$ | $E_{HAHA}$ | $E_{WSWS}$ | $jG_{0,\frac{1}{2}}$ | $jG_{0,\frac{1}{2}}$ | $-G_{0,1}^{-1}$ | $E_{HAWA}$ | $E_{HAWA}$ | $E_{WSHS}$ | $jG_{0,\frac{1}{2}}$ | $jG_{0,\frac{1}{2}}$ | $-G_{0,1}^{-1}$ |
| $E_{WSWA}$ | $E_{WSWA}$ | $E_{HSHS}$ | $G_{\frac{1}{2},0}$ | $G_{\frac{1}{2},0}$ | $G_{\frac{1}{2},0}^{-1}$ | $E_{WSHA}$ | $E_{WSHA}$ | $E_{HSWS}$ | $G_{\frac{1}{2},0}$ | $G_{\frac{1}{2},0}$ | $G_{\frac{1}{2},0}^{-1}$ |
| $E_{WSWA}$ | $E_{WAWS}$ | $E_{HAHA}$ | $G_{\frac{1}{2},0}$ | $jG_{\frac{1}{2},0}$ | $-jG_{\frac{1}{2},0}^{-1}$ | $E_{WSHA}$ | $E_{WAHS}$ | $E_{HAWA}$ | $G_{\frac{1}{2},0}$ | $jG_{\frac{1}{2},0}$ | $-jG_{\frac{1}{2},0}^{-1}$ |
| $E_{WAWS}$ | $E_{WAWS}$ | $E_{HSHS}$ | $jG_{\frac{1}{2},0}$ | $jG_{\frac{1}{2},0}$ | $-G_{\frac{1}{2},0}^{-1}$ | $E_{WAWS}$ | $E_{WAHS}$ | $E_{HSWS}$ | $jG_{\frac{1}{2},0}$ | $jG_{\frac{1}{2},0}$ | $-G_{\frac{1}{2},0}^{-1}$ |
| $E_{HSHA}$ | $E_{WSWA}$ | $E_{HSHA}$ | $G_{\frac{1}{2},\frac{1}{2}}$ | $G_{\frac{1}{2},0}$ | $G_{\frac{1}{2},\frac{1}{2}}^{-1}$ | $E_{HSWA}$ | $E_{WSHA}$ | $E_{HSWA}$ | $G_{\frac{1}{2},\frac{1}{2}}$ | $G_{\frac{1}{2},0}$ | $G_{\frac{1}{2},\frac{1}{2}}^{-1}$ |
| $E_{HSHA}$ | $E_{WAWS}$ | $E_{HAHS}$ | $G_{\frac{1}{2},\frac{1}{2}}$ | $jG_{\frac{1}{2},0}$ | $-jG_{\frac{1}{2},\frac{1}{2}}^{-1}$ | $E_{HSWA}$ | $E_{WAHS}$ | $E_{HAWS}$ | $G_{\frac{1}{2},\frac{1}{2}}$ | $jG_{\frac{1}{2},0}$ | $-jG_{\frac{1}{2},\frac{1}{2}}^{-1}$ |
| $E_{HAHS}$ | $E_{WSWA}$ | $E_{HAHS}$ | $jG_{\frac{1}{2},\frac{1}{2}}$ | $G_{\frac{1}{2},0}$ | $-jG_{\frac{1}{2},\frac{1}{2}}^{-1}$ | $E_{HAWS}$ | $E_{WSHA}$ | $E_{HAWS}$ | $jG_{\frac{1}{2},\frac{1}{2}}$ | $G_{\frac{1}{2},0}$ | $-jG_{\frac{1}{2},\frac{1}{2}}^{-1}$ |
| $E_{HAHS}$ | $E_{WAWS}$ | $E_{HSHA}$ | $jG_{\frac{1}{2},\frac{1}{2}}$ | $jG_{\frac{1}{2},0}$ | $-G_{\frac{1}{2},\frac{1}{2}}^{-1}$ | $E_{HAWS}$ | $E_{WAHS}$ | $E_{HSWA}$ | $jG_{\frac{1}{2},\frac{1}{2}}$ | $jG_{\frac{1}{2},0}$ | $-G_{\frac{1}{2},\frac{1}{2}}^{-1}$ |
| $E_{HSHA}$ | $E_{HSHA}$ | $E_{HSHS}$ | $G_{\frac{1}{2},\frac{1}{2}}$ | $G_{\frac{1}{2},\frac{1}{2}}$ | $G_{\frac{1}{2},1}^{-1}$ | $E_{HSWA}$ | $E_{HSWA}$ | $E_{HSWS}$ | $G_{\frac{1}{2},\frac{1}{2}}$ | $G_{\frac{1}{2},\frac{1}{2}}$ | $G_{\frac{1}{2},1}^{-1}$ |
| $E_{HSHA}$ | $E_{HAHS}$ | $E_{HAHS}$ | $G_{\frac{1}{2},\frac{1}{2}}$ | $jG_{\frac{1}{2},\frac{1}{2}}$ | $-jG_{\frac{1}{2},1}^{-1}$ | $E_{HSWA}$ | $E_{HAWS}$ | $E_{HAWA}$ | $G_{\frac{1}{2},\frac{1}{2}}$ | $jG_{\frac{1}{2},\frac{1}{2}}$ | $-jG_{\frac{1}{2},1}^{-1}$ |
| $E_{HAHS}$ | $E_{HAHS}$ | $E_{HSHS}$ | $jG_{\frac{1}{2},\frac{1}{2}}$ | $jG_{\frac{1}{2},\frac{1}{2}}$ | $-G_{\frac{1}{2},1}^{-1}$ | $E_{HAWS}$ | $E_{HAWS}$ | $E_{HSWS}$ | $jG_{\frac{1}{2},\frac{1}{2}}$ | $jG_{\frac{1}{2},\frac{1}{2}}$ | $-G_{\frac{1}{2},1}^{-1}$ |

rewritten as $\boldsymbol{H}_C = \boldsymbol{G}_{0,\frac{1}{2},M}^{-1}\boldsymbol{D}_M^{-1}\boldsymbol{\mathcal{H}}_F\boldsymbol{G}_{0,\frac{1}{2},M}$, where $\boldsymbol{\mathcal{H}}_F = \boldsymbol{D}_M \text{diag}\left\{\boldsymbol{G}_{0,0,M}\boldsymbol{E}_{WSWS}\boldsymbol{h}\right\}$. Equation (47) then becomes

$$
\begin{aligned}
\boldsymbol{d} &= \boldsymbol{R}_L \boldsymbol{G}_{0,\frac{1}{2},M}^{-1}\left[\operatorname{diag}\left\{\boldsymbol{G}_{0,0,M}\boldsymbol{E}_{WSWS}\boldsymbol{h}\right\}\boldsymbol{G}_{0,\frac{1}{2},M}\boldsymbol{E}_{HSHS}\theta\right] \\
&= \boldsymbol{R}_L \boldsymbol{G}_{0,\frac{1}{2},M}^{-1}\left[\boldsymbol{G}_{0,0,M}\boldsymbol{E}_{WSWS}\boldsymbol{h} \odot \boldsymbol{G}_{0,\frac{1}{2},M}\boldsymbol{E}_{HSHS}\theta\right] \\
&= \boldsymbol{R}_L \boldsymbol{G}_{0,\frac{1}{2},M}^{-1}\left[\widetilde{\boldsymbol{C}}_{1e,M}\boldsymbol{h} \odot \widetilde{\boldsymbol{C}}_{2e,M}\theta\right] \\
&= \boldsymbol{R}_L \boldsymbol{G}_{0,\frac{1}{2},M}^{-1}\boldsymbol{E}_{WSWA}\left[\boldsymbol{C}_{1e,N}\boldsymbol{h} \odot \boldsymbol{C}_{2e,N}\theta\right] \\
&= \boldsymbol{R}_L \widetilde{\boldsymbol{C}}_{2e,M}^{-1}\left[\boldsymbol{C}_{1e,N}\boldsymbol{h} \odot \boldsymbol{C}_{2e,N}\theta\right] \\
&= \boldsymbol{C}_{2e,N}^{-1}\left[\boldsymbol{C}_{1e,N}\boldsymbol{h} \odot \boldsymbol{C}_{2e,N}\theta\right].
\end{aligned} \tag{48}
$$

The matrix $\boldsymbol{R}_L$ retains the first $N$ samples of the $M \times 1$ vector $\widetilde{\boldsymbol{C}}_{2e,M}^{-1}\left[\boldsymbol{C}_{1e,N}\boldsymbol{h} \odot \boldsymbol{C}_{2e,N}\theta\right]$. An alternate way to express the result of Eq. (48) is in the matrix form

$$
\boldsymbol{d} = \boldsymbol{C}_{2e,N}^{-1}\mathcal{H}_{T,s}\boldsymbol{C}_{2e,N}\theta, \tag{49}
$$

where the $N \times N$ diagonal matrix $\mathcal{H}_{T,s} = \operatorname{diag}\left\{\boldsymbol{C}_{1e,N}\boldsymbol{h}\right\}$. The lowercase subscript *'s'* indicates the vector $\boldsymbol{C}_{1e,N}\boldsymbol{h}$ has whole-sample *symmetry* in the transform domain. The end results of Eqs. (48) or (49) are exactly equivalent to the same case of symmetric convolution expressed using operator notation in Table 3.

By making the appropriate substitutions from Table 5, the remaining 39 cases of symmetric convolution are just as easily derived. The vector-matrix derivation presented here is therefore equivalent to all forty cases of the symmetric convolution-multiplication property of the DTTs which Martucci [34] derives using operator notation.

### 3.2 *Extension to Multiple Dimensions*

An advantage to deriving the symmetric convolution-multiplication property in terms of vectors and matrices is that for orthogonally-sampled data in multiple dimensions, the results of the previous section extend naturally using Kronecker products as defined in Eq. (5). Data which are orthogonally sampled in two dimensions have their samples aligned on a unit-square grid. Before extending the results of the previous section, diagonalizing forms for the two-dimensional

odd DFT are presented. The two-dimensional odd DFT is used in cases where skew-circular convolution is the underlying form of convolution in symmetric convolution. These diagonalizing forms lay the groundwork for the presentation which follows on two-dimensional forms of the previous section's results. The case of multiple dimensions higher than two is also considered in this section.

A result similar to the diagonalizing form of two-dimensional circular convolution in Eq. (7) exists for two-dimensional skew-circular convolution based on the two-dimensional odd DFT. In this case $\boldsymbol{H}_{BS}$ is an $N_1N_2 \times N_1N_2$ *block skew-circulant* matrix of partitioned blocks of skew-circulant matrices arranged in a skew-circulant pattern. Odd DFT matrices will diagonalize the two-dimensional skew-circulant convolution matrix, $\boldsymbol{H}_{BS}$, through the relation

$$\boldsymbol{H}_{BS} = \left(\boldsymbol{V}_{N_1} \otimes \boldsymbol{V}_{N_2}\right) \mathscr{H}_O \left(\boldsymbol{V}_{N_1}^{-1} \otimes \boldsymbol{V}_{N_2}^{-1}\right). \tag{50}$$

The matrix $\mathscr{H}_O$ equals $\operatorname{diag}\left\{\left(\boldsymbol{V}_{N_1}^{-1} \otimes \boldsymbol{V}_{N_2}^{-1}\right)\boldsymbol{h}\right\}$, where $\boldsymbol{h} = \operatorname{vec}\{\boldsymbol{H}\}$, the lexicographic representation of a system point spread function (PSF), $\boldsymbol{H}$. Thus the two-dimensional skew-circular convolution-multiplication property is

$$\left(\boldsymbol{V}_{N_1}^{-1} \otimes \boldsymbol{V}_{N_2}^{-1}\right)\boldsymbol{d} = \left(\boldsymbol{V}_{N_1}^{-1} \otimes \boldsymbol{V}_{N_2}^{-1}\right)\boldsymbol{h} \odot \left(\boldsymbol{V}_{N_1}^{-1} \otimes \boldsymbol{V}_{N_2}^{-1}\right)\theta, \tag{51}$$

which is similar to Eq. (8).

The possibilities also exist for a matrix of partitioned blocks of circulant matrices arranged in a skew-circulant pattern, $\boldsymbol{H}_{S\times C}$, or a matrix of partitioned blocks of skew-circulant matrices arranged in a circulant pattern, $\boldsymbol{H}_{C\times S}$. The $N_1N_2 \times N_1N_2$ matrices $\boldsymbol{H}_{S\times C}$ and $\boldsymbol{H}_{C\times S}$ operate on lexicographically-ordered vectors which represent matrices. The subscript '$S \times C$' indicates that the matrix $\boldsymbol{H}_{S\times C}$ operates on the samples corresponding to the rows using circular convolution

and on the samples corresponding to the columns using skew-circular convolution. The converse is true for the matrix $\boldsymbol{H}_{C\times S}$. The diagonalizing forms of these particular cases respectively are

$$\boldsymbol{H}_{S\times C} = \left(\boldsymbol{V}_{N_1} \otimes \boldsymbol{W}_{N_2}\right) \mathscr{H}_{O\times F} \left(\boldsymbol{V}_{N_1}^{-1} \otimes \boldsymbol{W}_{N_2}^{-1}\right), \tag{52}$$

and

$$\boldsymbol{H}_{C\times S} = \left(\boldsymbol{W}_{N_1} \otimes \boldsymbol{V}_{N_2}\right) \mathscr{H}_{F\times O} \left(\boldsymbol{W}_{N_1}^{-1} \otimes \boldsymbol{V}_{N_2}^{-1}\right). \tag{53}$$

In Eqs. (52) and (53), the $N_1N_2 \times N_1N_2$ matrices $\mathscr{H}_{O\times F}$ and $\mathscr{H}_{F\times O}$ equal $\text{diag}\left\{\left(\boldsymbol{V}_{N_1}^{-1} \otimes \boldsymbol{W}_{N_2}^{-1}\right)\boldsymbol{h}\right\}$ and $\text{diag}\left\{\left(\boldsymbol{W}_{N_1}^{-1} \otimes \boldsymbol{V}_{N_2}^{-1}\right)\boldsymbol{h}\right\}$, respectively.

The problem of calculating the two-dimensional symmetric convolution of two $L_1 \times L_2$ matrices, $\boldsymbol{H}$ and $\boldsymbol{\Theta}$, can be made less computationally intense by the above diagonalizing forms of block circulant matrices, block skew-circulant matrices, or combinations of the two. The matrices $\boldsymbol{H}$ and $\boldsymbol{\Theta}$ have dimension $L_1 \times L_2$ because in general $L_1$ equals either $N_1 - 1$, $N_1$, or $N_1 + 1$ and $L_2$ equals either $N_2 - 1$, $N_2$, or $N_2 + 1$ based on the type of symmetric extension underlying the rows and columns of $\boldsymbol{H}$ and $\boldsymbol{\Theta}$.

An equivalent way to represent the matrices $\boldsymbol{H}$ and $\boldsymbol{\Theta}$, is by the $L_1L_2 \times 1$ lexicographically-ordered vectors $\boldsymbol{h}$ and $\theta$. The two-dimensional form of Eq. (41) is thus

$$\boldsymbol{d} = \left(\boldsymbol{R}_{L_1} \otimes \boldsymbol{R}_{L_2}\right) \boldsymbol{H}_B \left(\boldsymbol{E}_{1c} \otimes \boldsymbol{E}_{1r}\right) \theta. \tag{54}$$

The matrices $\boldsymbol{R}_{L_1}$ and $\boldsymbol{R}_{L_2}$ have the same form as $\boldsymbol{R}_L$ from before, which is $\boldsymbol{R}_{L_1} = \left[\boldsymbol{I}_{L_1} \;\; \boldsymbol{0}\right]$ and $\boldsymbol{R}_{L_2} = \left[\boldsymbol{I}_{L_2} \;\; \boldsymbol{0}\right]$. The $L_1L_2 \times M_1M_2$ windowing matrix $\left(\boldsymbol{R}_{L_1} \otimes \boldsymbol{R}_{L_2}\right)$ thus retains $L_1$ samples from each column and $L_2$ samples from each row. The symmetric extension matrices $\boldsymbol{E}_{1c}$ and $\boldsymbol{E}_{1r}$ operate on the rows and columns of $\boldsymbol{\Theta}$ separately giving it possibly different symmetry in horizontal and vertical directions. The dimensions of $\boldsymbol{E}_{1c}$ and $\boldsymbol{E}_{1r}$ will be $M_1 \times L_1$ and $M_2 \times L_2$, re-

spectively, where $M_1 = 2N_1$ for $N_1$ even or $M_1 = 2N_1 - 1$ for $N_1$ odd, and $M_2 = 2N_2$ for $N_2$ even or $M_2 = 2N_2 - 1$ for $N_2$ odd. The $M_1M_2 \times M_1M_2$ matrix $\boldsymbol{H}_B$ can be either block circulant, block skew-circulant, or a combination depending on the underlying symmetry of the rows and columns of the $L_1 \times L_2$ matrix $\boldsymbol{H}$. The matrix $\boldsymbol{H}_B$ will have the vector $(\boldsymbol{E}_{2c} \otimes \boldsymbol{E}_{2r})\boldsymbol{h}$ as its first column. The matrices $\boldsymbol{E}_{2c}$ and $\boldsymbol{E}_{2r}$ will also have dimension $M_1 \times L_1$ and $M_2 \times L_2$, respectively. They will perform a two-dimensional symmetric extension on the matrix $\boldsymbol{H}$, just as the matrices $\boldsymbol{E}_{1c}$ and $\boldsymbol{E}_{1r}$ perform a two-dimensional symmetric extension on the matrix $\boldsymbol{\Theta}$.

As before in the one-dimensional case, all of the symmetric extensions, shifts, additions, and multiplications of two-dimensional symmetric convolution in Eq. (54) can be contained in a single $L_1L_2 \times L_1L_2$ matrix $\boldsymbol{H}_{BSC} = (\boldsymbol{R}_{L_1} \otimes \boldsymbol{R}_{L_2})\boldsymbol{H}_B(\boldsymbol{E}_{1c} \otimes \boldsymbol{E}_{1r})$, so that $\boldsymbol{d} = \boldsymbol{H}_{BSC}\theta$. The subscript *'BSC'* denotes *block symmetric convolution*. It also follows from the one-dimensional case that a diagonal form, $\mathcal{H}_T$, exists for the matrix $\boldsymbol{H}_{BSC}$. The diagonalization in two dimensions is carried out using Kronecker products of the DTTs which have the form $(\boldsymbol{T}_{c,L_1} \otimes \boldsymbol{T}_{r,L_2})$. The $L_1 \times L_1$ DTT matrix $\boldsymbol{T}_{c,L_1}$ acts on the samples of a lexicographic vector which represent the columns of a matrix. Similarly, the $L_2 \times L_2$ DTT matrix $\boldsymbol{T}_{r,L_2}$ acts on the samples which represent the rows.

A summary of the two-dimensional transforms discussed thus far appears in Table 6. All of the matrices in Table 6 related to the DFT or odd DFT have dimension $M_1M_2 \times M_1M_2$ because they act on symmetric extensions of matrices inside the definition of symmetric convolution. The two-dimensional trigonometric transform matrix, $(\boldsymbol{T}_{c,L_1} \otimes \boldsymbol{T}_{r,L_2})$, has dimension $L_1L_2 \times L_1L_2$ because it acts directly on a lexicographic vector representing a matrix. The matrices $\boldsymbol{H}_{BSC}$ and $\mathcal{H}_T$ in Table 6 also have dimension $L_1L_2 \times L_1L_2$.

Table 6. Two-Dimensional Transforms

| Type | Matrix Representation | Underlying Form of Convolution | Convolution Matrix | Diagonal Form in Transform Domain |
|---|---|---|---|---|
| DFT on rows and columns | $\left(\boldsymbol{W}_{M_1}^{-1} \otimes \boldsymbol{W}_{M_2}^{-1}\right)$ | circular for rows and columns | $\boldsymbol{H}_{BC}$ | $\mathcal{H}_F$ |
| odd DFT on rows and columns | $\left(\boldsymbol{V}_{M_1}^{-1} \otimes \boldsymbol{V}_{M_2}^{-1}\right)$ | skew-circular for rows and columns | $\boldsymbol{H}_{BS}$ | $\mathcal{H}_O$ |
| DFT on rows; odd DFT on columns | $\left(\boldsymbol{V}_{M_1}^{-1} \otimes \boldsymbol{W}_{M_2}^{-1}\right)$ | circular for rows; skew-circular for columns | $\boldsymbol{H}_{S\times C}$ | $\mathcal{H}_{O\times F}$ |
| odd DFT on rows; DFT on columns | $\left(\boldsymbol{W}_{M_1}^{-1} \otimes \boldsymbol{V}_{M_2}^{-1}\right)$ | skew-circular for rows; circular for columns | $\boldsymbol{H}_{C\times S}$ | $\mathcal{H}_{F\times O}$ |
| DTT on rows and columns | $\left(\boldsymbol{T}_{c,L_1} \otimes \boldsymbol{T}_{r,L_2}\right)$ | symmetric for rows and columns | $\boldsymbol{H}_{BSC}$ | $\mathcal{H}_T$ |

The procedure to extend symmetric convolution to multiple dimensions follows a similar derivation as the previous section's derivation for one dimension. Here, substitutions from Eqs. (34), (35), (39), and (40) are made into the appropriate diagonal form in Eqs. (6), (50), (52), or (53). The DTT matrices which result from these substitutions appear in the two-dimensional equivalent of Eq. (45) as

$$\boldsymbol{d} = \left(\boldsymbol{T}_{3c}^{-1} \otimes \boldsymbol{T}_{3r}^{-1}\right)\left[\left(\boldsymbol{T}_{2c} \otimes \boldsymbol{T}_{2r}\right)\boldsymbol{h} \odot \left(\boldsymbol{T}_{1c} \otimes \boldsymbol{T}_{1r}\right)\boldsymbol{\theta}\right]. \tag{55}$$

An alternate way to express Eq. (55) is

$$\boldsymbol{D} = \boldsymbol{T}_{3c}^{-1}\left[\left(\boldsymbol{T}_{2c}\boldsymbol{H}\boldsymbol{T}_{2r}^{T}\right) \odot \left(\boldsymbol{T}_{1c}\boldsymbol{\Theta}\boldsymbol{T}_{1r}^{T}\right)\right]\boldsymbol{T}_{3r}^{-T}. \tag{56}$$

The two results in Eqs. (55) and (56) are equivalent because for any separable transform represented by the matrix $\boldsymbol{U}$, the vector $\boldsymbol{x}_U = \left(\boldsymbol{U}_{N_1} \otimes \boldsymbol{U}_{N_2}\right)\boldsymbol{x}$ is lexicographically equivalent to the matrix $\boldsymbol{\mathcal{X}}_U = \boldsymbol{U}_{N_1}\boldsymbol{X}\boldsymbol{U}_{N_2}^{T}$ [27]. The DFT, the odd DFT, and the entire family of trigonometric transforms are all separable. In Eqs. (55) and (56) the set of DTTs, $\left\{\boldsymbol{T}_{1r},\ \boldsymbol{T}_{2r},\ \boldsymbol{T}_{3r}^{-1}\right\}$, which acts on the rows and the set of DTTs, $\left\{\boldsymbol{T}_{1c},\ \boldsymbol{T}_{2c},\ \boldsymbol{T}_{3c}^{-1}\right\}$, which acts on the columns must each come from one of the allowable sets of forty transforms in Table 3, but they need not be the same set. They may

be different because the matrices $\boldsymbol{H}$ and $\boldsymbol{\Theta}$ may have different underlying symmetry in each direction.

These same diagonalizing principles apply to higher dimensional data by cascading the Kronecker products in the diagonalizing equations. For example a $D$-dimensional circulant convolution matrix acting on $D$-dimensional data of size $N_1 \times N_2 \times \cdots \times N_D$ is diagonalized by

$$\boldsymbol{H}_{C\times C\times\cdots\times C} = \left(\boldsymbol{W}_{N_1} \otimes \boldsymbol{W}_{N_2} \otimes \cdots \otimes \boldsymbol{W}_{N_D}\right) \boldsymbol{\mathcal{H}}_F \left(\boldsymbol{W}_{N_1}^{-1} \otimes \boldsymbol{W}_{N_2}^{-1} \otimes \cdots \otimes \boldsymbol{W}_{N_D}^{-1}\right), \tag{57}$$

where $\boldsymbol{\mathcal{H}}_F$ is diagonal with the vector $\left(\boldsymbol{W}_{N_1}^{-1} \otimes \boldsymbol{W}_{N_2}^{-1} \otimes \cdots \otimes \boldsymbol{W}_{N_D}^{-1}\right)\boldsymbol{h}$ along the diagonal. In this $D$-dimensional case, there are $2^D$ possible combinations of circulant and skew-circulant multiple sub-blocks in the block circulant structure of $\boldsymbol{H}_B$. Also in this multiple dimensional case, the points of symmetry underlying the sequences in one dimension and the lines of symmetry underlying the sequences in two dimensions become planes of symmetry in three dimensions and hyperplanes of symmetry in more than three dimensions. In the next section, the behavior of symmetric convolution is examined for cases where the sequences to be convolved do not possess the underlying symmetry required for the symmetric convolution-multiplication property of the DTTs to hold.

### 3.3 *Extension to Asymmetric Sequences*

Even though the symmetric convolution-multiplication property of the DTTs extends easily to multiple dimensions, it is still limited by the underlying symmetry in the sequences to be convolved or multiplied in the transform domain. For multidimensional sequences, there must be symmetry in every dimension [34]. Martucci mentions [34] that asymmetric sequences can be decomposed into their symmetric and antisymmetric parts, transformed using different transforms because of the different underlying symmetry, and then multiplied.

This decomposition is straightforward for one-dimension, but can be complicated for two or more dimensions. A finite one-dimensional sequence, $x(n)$, which is zero outside the region $0 \le n \le N-1$, can be decomposed as $x(n) = x_a(n) + x_s(n)$ in two ways. Using whole-sample symmetry,

$$x_a(n) = \begin{cases} -\frac{1}{2}x(-n), & n < 0 \\ 0, & n = 0 \\ \frac{1}{2}x(n), & n > 0, \end{cases} \quad \text{and} \quad x_s(n) = \begin{cases} \frac{1}{2}x(-n), & n < 0 \\ x(n), & n = 0 \\ \frac{1}{2}x(n), & n > 0, \end{cases} \tag{58}$$

where $x_a(n)$ has WA symmetry to the left and $x_s(n)$ has WS symmetry to the left. The symmetry to the right for $x_a(n)$ can be any one of the four types (WA, WS, HA, or HS) as long as the symmetric extension of $x_s(n)$ to the right cancels with it to yield $x(n) = x_a(n) + x_s(n)$ which is zero outside the region $0 \le n \le N-1$. For example if $x_a(n)$ has WAHS symmetry, then $x_s(n)$ must have WSHA symmetry. Similarly using half-sample symmetry to the left,

$$x_a(n) = \begin{cases} -\frac{1}{2}x(-n-1), & n < 0 \\ \frac{1}{2}x(n), & n \ge 0, \end{cases} \quad \text{and} \quad x_s(n) = \begin{cases} \frac{1}{2}x(-n-1), & n < 0 \\ \frac{1}{2}x(n), & n \ge 0. \end{cases} \tag{59}$$

Again the right-hand symmetry is arbitrary as long as the two types of right-hand symmetry for $x_a(n)$ and $x_s(n)$ cancel with each other for the two sequences.

In two dimensions, there are four ways to decompose an orthogonally-sampled sequence using combinations of half-sample and whole-sample symmetry. Each decomposition must have four terms consisting of symmetric and antisymmetric parts in each dimension so that

$$x(n_1,n_2) = x_{aa}(n_1,n_2) + x_{as}(n_1,n_2) + x_{sa}(n_1,n_2) + x_{ss}(n_1,n_2). \tag{60}$$

Definitions of the four terms for each of the four possible decompositions appear in Table 7. The orthogonal coordinate system axes form the lines of symmetry because of orthogonal sampling. These definitions exist for a sequence in the first quadrant, *i.e.*, $0 \le n_1 \le N_1 - 1$, $0 \le n_2 \le N_2 - 1$.

Table 7. Decomposition of a Two-Dimensional Asymmetric Sequence into Symmetric and Antisymmetric Parts

| | $n_1$ whole - sample symmetry, $n_2$ whole - sample symmetry | | | |
|---|---|---|---|---|
| | $x_{aa}(n_1,n_2)$ | $x_{as}(n_1,n_2)$ | $x_{sa}(n_1,n_2)$ | $x_{ss}(n_1,n_2)$ |
| $n_1<0,n_2<0$ | $\frac{1}{4}x(-n_1,-n_2)$ | $-\frac{1}{4}x(-n_1,-n_2)$ | $-\frac{1}{4}x(-n_1,-n_2)$ | $\frac{1}{4}x(-n_1,-n_2)$ |
| $n_1<0,n_2=0$ | 0 | $-\frac{1}{2}x(-n_1,0)$ | 0 | $\frac{1}{2}x(-n_1,0)$ |
| $n_1<0,n_2>0$ | $-\frac{1}{4}x(-n_1,n_2)$ | $-\frac{1}{4}x(-n_1,n_2)$ | $\frac{1}{4}x(-n_1,n_2)$ | $\frac{1}{4}x(-n_1,n_2)$ |
| $n_1=0,n_2<0$ | 0 | 0 | $-\frac{1}{2}x(0,-n_2)$ | $\frac{1}{2}x(0,-n_2)$ |
| $n_1=n_2=0$ | 0 | 0 | 0 | $x(0,0)$ |
| $n_1=0,n_2>0$ | 0 | 0 | $\frac{1}{2}x(0,n_2)$ | $\frac{1}{2}x(0,n_2)$ |
| $n_1>0,n_2<0$ | $-\frac{1}{4}x(n_1,-n_2)$ | $\frac{1}{4}x(n_1,-n_2)$ | $-\frac{1}{4}x(n_1,-n_2)$ | $\frac{1}{4}x(n_1,-n_2)$ |
| $n_1>0,n_2=0$ | 0 | $\frac{1}{2}x(n_1,0)$ | 0 | $\frac{1}{2}x(n_1,0)$ |
| $n_1>0,n_2>0$ | $\frac{1}{4}x(n_1,n_2)$ | $\frac{1}{4}x(n_1,n_2)$ | $\frac{1}{4}x(n_1,n_2)$ | $\frac{1}{4}x(n_1,n_2)$ |
| | $n_1$ whole - sample symmetry, $n_2$ half - sample symmetry | | | |
| | $x_{aa}(n_1,n_2)$ | $x_{as}(n_1,n_2)$ | $x_{sa}(n_1,n_2)$ | $x_{ss}(n_1,n_2)$ |
| $n_1<0,n_2<0$ | $\frac{1}{4}x(-n_1,-n_2-1)$ | $-\frac{1}{4}x(-n_1,-n_2-1)$ | $-\frac{1}{4}x(-n_1,-n_2-1)$ | $\frac{1}{4}x(-n_1,-n_2-1)$ |
| $n_1<0,n_2\geq 0$ | $-\frac{1}{4}x(-n_1,n_2)$ | $-\frac{1}{4}x(-n_1,n_2)$ | $\frac{1}{4}x(-n_1,n_2)$ | $\frac{1}{4}x(-n_1,n_2)$ |
| $n_1=0,n_2<0$ | 0 | 0 | $-\frac{1}{2}x(0,-n_2-1)$ | $\frac{1}{2}x(0,-n_2-1)$ |
| $n_1=0,n_2\geq 0$ | 0 | 0 | $\frac{1}{2}x(0,n_2)$ | $\frac{1}{2}x(0,n_2)$ |
| $n_1>0,n_2<0$ | $-\frac{1}{4}x(n_1,-n_2-1)$ | $\frac{1}{4}x(n_1,-n_2-1)$ | $-\frac{1}{4}x(n_1,-n_2-1)$ | $\frac{1}{4}x(n_1,-n_2-1)$ |
| $n_1>0,n_2\geq 0$ | $\frac{1}{4}x(n_1,n_2)$ | $\frac{1}{4}x(n_1,n_2)$ | $\frac{1}{4}x(n_1,n_2)$ | $\frac{1}{4}x(n_1,n_2)$ |
| | $n_1$ half - sample symmetry, $n_2$ whole - sample symmetry | | | |
| | $x_{aa}(n_1,n_2)$ | $x_{as}(n_1,n_2)$ | $x_{sa}(n_1,n_2)$ | $x_{ss}(n_1,n_2)$ |
| $n_1<0,n_2<0$ | $\frac{1}{4}x(-n_1-1,-n_2)$ | $-\frac{1}{4}x(-n_1-1,-n_2)$ | $-\frac{1}{4}x(-n_1-1,-n_2)$ | $\frac{1}{4}x(-n_1-1,-n_2)$ |
| $n_1<0,n_2=0$ | 0 | $-\frac{1}{2}x(-n_1-1,0)$ | 0 | $\frac{1}{2}x(-n_1-1,0)$ |
| $n_1<0,n_2>0$ | $-\frac{1}{4}x(-n_1-1,n_2)$ | $-\frac{1}{4}x(-n_1-1,n_2)$ | $\frac{1}{4}x(-n_1-1,n_2)$ | $\frac{1}{4}x(-n_1-1,n_2)$ |
| $n_1\geq 0,n_2<0$ | $-\frac{1}{4}x(n_1,-n_2)$ | $\frac{1}{4}x(n_1,-n_2)$ | $-\frac{1}{4}x(n_1,-n_2)$ | $\frac{1}{4}x(n_1,-n_2)$ |
| $n_1\geq 0,n_2=0$ | 0 | $\frac{1}{2}x(n_1,0)$ | 0 | $\frac{1}{2}x(n_1,0)$ |
| $n_1\geq 0,n_2>0$ | $\frac{1}{4}x(n_1,n_2)$ | $\frac{1}{4}x(n_1,n_2)$ | $\frac{1}{4}x(n_1,n_2)$ | $\frac{1}{4}x(n_1,n_2)$ |
| | $n_1$ half - sample symmetry, $n_2$ half - sample symmetry | | | |
| | $x_{aa}(n_1,n_2)$ | $x_{as}(n_1,n_2)$ | $x_{sa}(n_1,n_2)$ | $x_{ss}(n_1,n_2)$ |
| $n_1<0,n_2<0$ | $\frac{1}{4}x(-n_1-1,-n_2-1)$ | $-\frac{1}{4}x(-n_1-1,-n_2-1)$ | $-\frac{1}{4}x(-n_1-1,-n_2-1)$ | $\frac{1}{4}x(-n_1-1,-n_2-1)$ |
| $n_1<0,n_2\geq 0$ | $-\frac{1}{4}x(-n_1-1,n_2)$ | $-\frac{1}{4}x(-n_1-1,n_2)$ | $\frac{1}{4}x(-n_1-1,n_2)$ | $\frac{1}{4}x(-n_1-1,n_2)$ |
| $n_1\geq 0,n_2<0$ | $-\frac{1}{4}x(n_1,-n_2-1)$ | $\frac{1}{4}x(n_1,-n_2-1)$ | $-\frac{1}{4}x(n_1,-n_2-1)$ | $\frac{1}{4}x(n_1,-n_2-1)$ |
| $n_1\geq 0,n_2\geq 0$ | $\frac{1}{4}x(n_1,n_2)$ | $\frac{1}{4}x(n_1,n_2)$ | $\frac{1}{4}x(n_1,n_2)$ | $\frac{1}{4}x(n_1,n_2)$ |

Similar definitions exist for sequences in other quadrants. This decomposition still applies to sequences which exist in more than one quadrant. The decomposition of a multiquadrant se-

quence requires it to be separated into subsequences which exist in the individual quadrants of the $n_1 - n_2$ plane. The four subsequences must then be individually decomposed into their anti-symmetric-antisymmetric, antisymmetric-symmetric, symmetric-antisymmetric and symmetric-symmetric parts, and the results added.

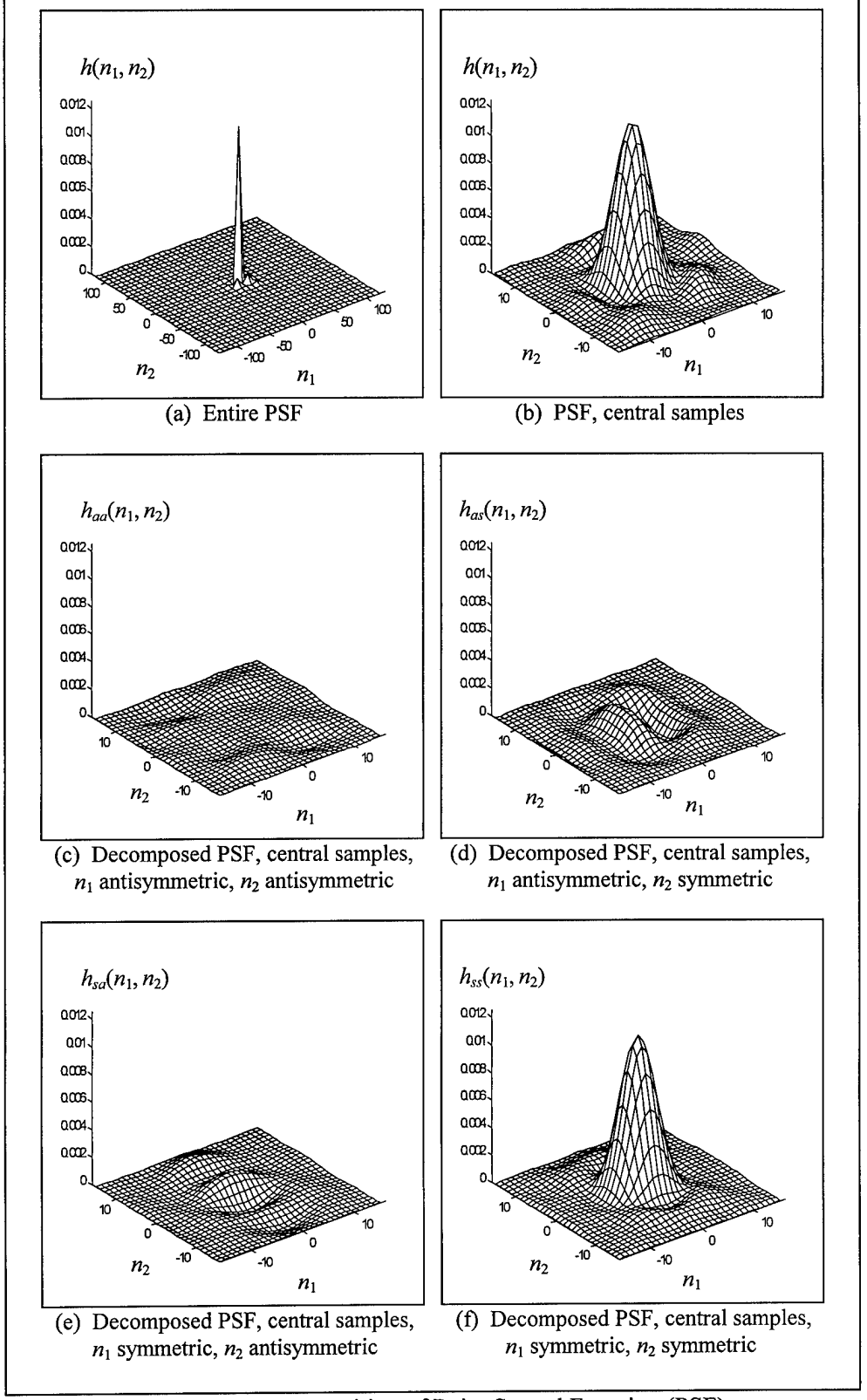
These concepts apply to orthogonally-sampled asymmetric sequences in three and higher dimensions as well. A three-dimensional sequence must be decomposed into eight symmetric and antisymmetric parts in each dimension. In general a $D$-dimensional sequence will have $2^D$ components in its decomposition. A three-dimensional sequence will have planes of symmetry, and a $D$-dimensional sequence will have hyperplanes of symmetry [34]. This multidimensional decomposition is very similar to the decomposition needed to implement a multidimensional Hilbert transform [19], except here the phase is reversed by 180° instead of being rotated by 90°.

In the previous discussion an alternate way to derive Martucci's symmetric convolution-multiplication property [34] was shown using vector-matrix notation and the property was extended to the more general class of asymmetric multidimensional sequences. The focus now turns to filtering asymmetric multidimensional sequences in the trigonometric transform domain.

### 3.4 *A Filtering Example*

A problem that requires the convolution of multidimensional asymmetric sequences is modeling the effects of atmospheric turbulence on imaging systems [40]. This model provides a good example of a problem which requires multidimensional asymmetry to demonstrate the symmetric convolution-multiplication property for DTTs because models for atmospheric turbulence result in asymmetric point spread functions (PSFs).

Figure 3 shows an example of a $256 \times 256$ PSF selected to model the effects of atmospheric turbulence on the aperture of an imaging system. The entire PSF appears in Figure 3(a).




(a) Entire PSF

(b) PSF, central samples

(c) Decomposed PSF, central samples, $n_1$ antisymmetric, $n_2$ antisymmetric

(d) Decomposed PSF, central samples, $n_1$ antisymmetric, $n_2$ symmetric

(e) Decomposed PSF, central samples, $n_1$ symmetric, $n_2$ antisymmetric

(f) Decomposed PSF, central samples, $n_1$ symmetric, $n_2$ symmetric

Figure 3. Decomposition of Point Spread Function (PSF)

The central $32 \times 32$ samples appear in Figure 3(b) where the asymmetry is clearly evident. A direct transformation of this PSF into the trigonometric transform domain is not possible because of its asymmetry. The trigonometric transforms can only act directly on the decomposed symmetric and antisymmetric parts of the PSF about both the $n_1$ and $n_2$ axes. If $h(n_1,n_2)$ denotes the PSF, the decomposition becomes

$$h(n_1,n_2) = h_{aa}(n_1,n_2) + h_{as}(n_1,n_2) + h_{sa}(n_1,n_2) + h_{ss}(n_1,n_2). \tag{61}$$

The decomposition in Eq. (61) is not direct since $h(n_1,n_2)$ exists in all four quadrants of the $n_1 - n_2$ plane. The correct decomposition technique is to decompose the portion of the PSF in each quadrant separately and then add the results having like symmetry. For example, the term $h_{as}(n_1,n_2)$ in Eq. (61) which is antisymmetric in $n_1$ and symmetric in $n_2$ would arise from the sum of the antisymmetric-symmetric portions of the whole-sample symmetric decompositions for each quadrant. The central samples of the decomposition of this PSF using whole-sample symmetry for both $n_1$ and $n_2$ appear in Figures 3(c)-(f). The antisymmetric-antisymmetric portion of the decomposition, $h_{aa}(n_1,n_2)$, appears in Figure 3(c), $h_{as}(n_1,n_2)$ appears in Figure 3(d), $h_{sa}(n_1,n_2)$ appears in Figure 3(e), and $h_{ss}(n_1,n_2)$ appears in Figure 3(f). Half-sample symmetry for either or both directions could have just as easily been chosen for this decomposition instead of whole-sample symmetry. The only impact to this example would be the choice of trigonometric transform to apply later.

Figure 4 shows the negative of a $256 \times 256$ pixel computer-generated rendering of an ocean reconnaissance satellite. The goal of this filtering example is to convolve this satellite object with the PSF in Figure 3 by converting each to the trigonometric transform domain, point-multiplying the results and then inverse transforming the product back to the spatial domain. Decomposing the object to be imaged into its symmetric and antisymmetric parts is unnecessary

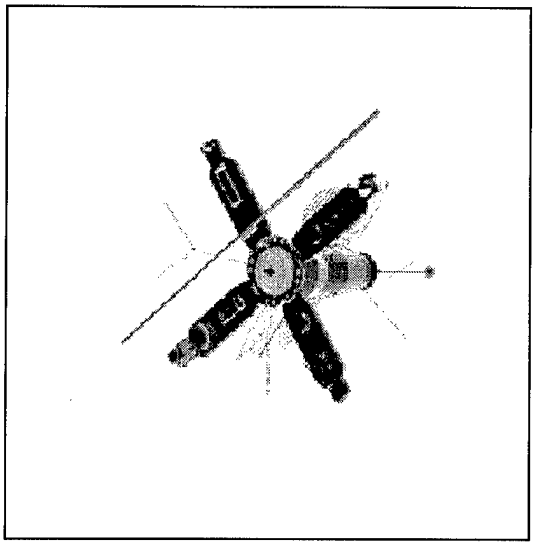


Figure 4. Simulated Satellite Object
(Negative Shown for Clarity)

because the convolution shift property [37] allows for a shift of the object's origin. The convolution shift property states that if $d_1(n) = h(n) * \theta(n)$ and $d_2(n) = h(n) * \theta(n - n_0)$, then $d_2(n) = d_1(n - n_0)$. The shift of the object's origin allows the entire object to appear as if it were one fundamental period of a two-dimensional symmetric periodic sequence. The symmetric periodic extensions of the object are external to the object itself and therefore transparent to the problem. The symmetric periodic extensions of the PSF must, however, occur internally within the subsequences $h_{aa}(n_1,n_2)$, $h_{as}(n_1,n_2)$, $h_{sa}(n_1,n_2)$, and $h_{ss}(n_1,n_2)$, and thus need to be accounted for.

The DTTs imply an underlying symmetric periodicity in both the sequence and the transform domains in the same way that the DFT of a finite sequence implies periodicity in both domains. Therefore the implementation of a filter in the transform domain does not require the full symmetric extensions of either the object or the PSF. The symmetry is implied by the trigonometric transforms just like the DFT implies periodicity. In fact since the symmetry of the four parts of the decomposition of the PSF must be accounted for internally for each subsequence, the filter needs only to retain the principal $128 \times 128$ values in the first quadrant of each part. These

parts in the first quadrant are designated by $h_{aa}^{rr}(n_1,n_2)$, $h_{as}^{rr}(n_1,n_2)$, $h_{sa}^{rr}(n_1,n_2)$, and $h_{ss}^{rr}(n_1,n_2)$, where the superscript *'rr'* indicates the filter right-half terms about both $n_1$ and $n_2$. The $128 \times 128$ right-half subsequences have the remainder of each array padded with zeros so their full sizes are all $256 \times 256$. This process is the two-dimensional equivalent of retaining just the right-half samples of the filter impulse response and then zero-padding in the one-dimensional examples of [34] and [35].

With all underlying symmetry properly accounted for, trigonometric transforms must next be applied to the object and the decomposed PSF to perform symmetric convolution via multiplication in the transform domain. In this example the finite length of the object being imaged is even in both directions, *i.e.*, $N_1 = N_2 = 256$, so any one of the eight even-length DTTs can be applied for each direction, $n_1$ and $n_2$. The same transform does not need to be applied for each direction. For this example let the matrix $\mathcal{O}$ represent the satellite object, and then apply a type-II DCT to both the rows and columns of $\mathcal{O}$. The transform domain representation of the matrix $\mathcal{O}$ thus becomes $\mathcal{O}_T = \boldsymbol{C}_{2e,N_1} \mathcal{O} \boldsymbol{C}_{2e,N_2}^T$. All matrices in this expression have dimension $256 \times 256$.

Thus far the filtering problem has allowed freedom to choose any type of symmetry in the decomposition of $h(n_1,n_2)$. There has also been freedom to choose any of the even-length transforms to apply to the object being filtered. In this example, whole-sample symmetry has been chosen in each direction for the decomposition of $h(n_1,n_2)$, and a type-II DCT has been chosen to transform both the rows and columns of the object. These two choices now dictate the type of transforms to apply to the individual components of the decomposition of the PSF, $h(n_1,n_2)$. The choices also restrict which inverse transforms can be used to produce the convolved image. From Table 3 which specifies the types of symmetric convolution, there are only two allowable even-length cases of symmetric convolution which have a type-II DCT operating on one se-

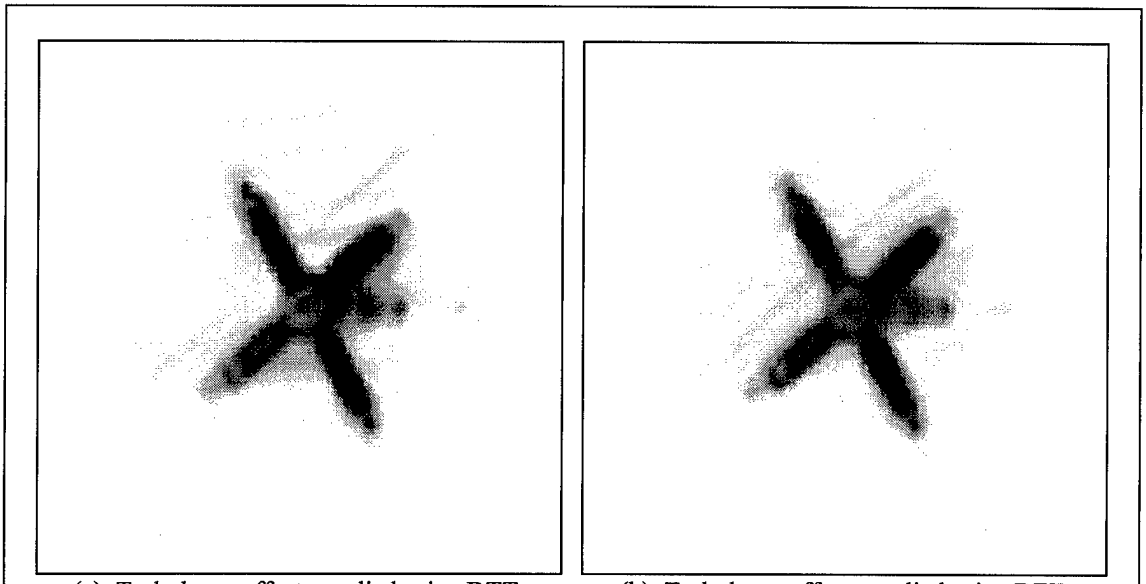
quence and whole-sample symmetry in the other sequence. In vector-matrix form, the allowable cases are $\boldsymbol{C}_{2e,N}^{-1}\left[\boldsymbol{C}_{1e,N}\boldsymbol{h} \odot \boldsymbol{C}_{2e,N}\theta\right]$ for $\boldsymbol{h}$ whole-sample symmetric, and $\boldsymbol{S}_{2e,N}^{-1}\left[\boldsymbol{S}_{1e,N}\boldsymbol{h} \odot \boldsymbol{C}_{2e,N}\theta\right]$ for $\boldsymbol{h}$ whole-sample antisymmetric. Thus the filter needs to apply a type-I DCT and a type-I DST to the appropriate symmetric and antisymmetric rows and columns of the decomposed subsequences of $h(n_1,n_2)$. If $\boldsymbol{H}_{aa}^{rr}$, $\boldsymbol{H}_{as}^{rr}$, $\boldsymbol{H}_{sa}^{rr}$, and $\boldsymbol{H}_{ss}^{rr}$ represent the $128\times128$ principal values of $h_{aa}(n_1,n_2)$, $h_{as}(n_1,n_2)$, $h_{sa}(n_1,n_2)$, and $h_{ss}(n_1,n_2)$ zero-padded to dimension $256\times256$, then the transforms to apply are

$$\begin{aligned}\mathcal{H}_{T,aa}^{rr} &= \boldsymbol{S}_{1e,N_1}\boldsymbol{H}_{aa}^{rr}\boldsymbol{S}_{1e,N_2}^{T}, \qquad \mathcal{H}_{T,as}^{rr} = \boldsymbol{C}_{1e,N_1}\boldsymbol{H}_{as}^{rr}\boldsymbol{S}_{1e,N_2}^{T},\\ \mathcal{H}_{T,sa}^{rr} &= \boldsymbol{S}_{1e,N_1}\boldsymbol{H}_{sa}^{rr}\boldsymbol{C}_{1e,N_2}^{T}, \text{ and } \mathcal{H}_{T,ss}^{rr} = \boldsymbol{C}_{1e,N_1}\boldsymbol{H}_{ss}^{rr}\boldsymbol{C}_{1e,N_2}^{T}.\end{aligned} \tag{62}$$

Each result in Eq. (62) must then be point-multiplied with $\mathcal{D}_T$ before inverting to return to the sequence domain. The final convolved result represented by the matrix $\boldsymbol{D}$ is thus expressible as the sum

$$\begin{aligned}\boldsymbol{D} = {} & \boldsymbol{S}_{2e,N_1}^{-1}\left[\mathcal{H}_{T,aa}^{rr} \odot \mathcal{D}_T\right]\boldsymbol{S}_{2e,N_2}^{-T} + \boldsymbol{C}_{2e,N_1}^{-1}\left[\mathcal{H}_{T,as}^{rr} \odot \mathcal{D}_T\right]\boldsymbol{S}_{2e,N_2}^{-T}\\ & + \boldsymbol{S}_{2e,N_1}^{-1}\left[\mathcal{H}_{T,sa}^{rr} \odot \mathcal{D}_T\right]\boldsymbol{C}_{2e,N_2}^{-T} + \boldsymbol{C}_{2e,N_1}^{-1}\left[\mathcal{H}_{T,ss}^{rr} \odot \mathcal{D}_T\right]\boldsymbol{C}_{2e,N_2}^{-T}.\end{aligned} \tag{63}$$

Figure 5(a) depicts the resulting image from Eq. (63). Figure 5(b) depicts the image resulting from a conventional DFT multiplication to implement circular convolution. Note the similarity between the two images and the effects of blurring caused by atmospheric turbulence in each. The average absolute difference between the pixels in Figures 5(a) and (b) is 0.024 gray levels, which is 0.0094% of the total dynamic range of 256 gray levels. The maximum difference between any two pixels is 0.270 gray levels, which is 0.11% of the dynamic range. The very slight differences between the images is caused by the very small outer values in the PSF wrapping back into the image in a slightly different manner for circular and symmetric convolution.



(a) Turbulence effects applied using DTTs to implement symmetric convolution.

(b) Turbulence effects applied using DFT to implement circular convolution.

Figure 5. Degraded Satellite Image Showing the Effects of Atmospheric Turbulence on a 1 m Circular Aperture (Negative Shown for Clarity)

To summarize the procedure used in this example and to implement the symmetric convolution-multiplication property in general, the following list of steps is provided:

(1) Decompose the filter impulse response into parts having support only in a single quadrant (or orthant, if $D > 2$).

(2) Decompose the parts existing in a single quadrant (or orthant) into antisymmetric and symmetric parts about each axis.

(3) Add parts with like symmetry from each decomposition, retain only the first-quadrant values, and pad with zeros.

(4) Apply a trigonometric transform to each dimension of the data.

(5) Select appropriate transform relations from Table 3 based on the type of symmetry selected in step (1) and the transforms selected in step (4).

(6) Apply the forward transforms to the impulse response matrices calculated in step (3).

(7) Point-multiply each part of the decomposed and transformed impulse response with the transformed data from step (4).

(8) Apply the inverse transforms determined from step (5) to each point-multiplied transform-domain sequence.

(9) Add the results of the inverse transforms to yield the symmetrically convolved sequence.

In this chapter, the results of deriving each of the forty forms of the symmetric convolution-multiplication property for discrete trigonometric transforms has been presented by showing how the transforms diagonalize a matrix which represents the symmetric convolution operation. Derived in this manner, the symmetric convolution-multiplication property extends easily to multiple dimensions. The filtering of multidimensional asymmetric sequences is then possible because symmetric convolution is equivalent to multiplication in the transform domain for each of the underlying types of symmetry in an asymmetric image. The next step in this development is to seek filters in the trigonometric transform domain which remove the effects of blurring caused by PSFs which are similar to the PSF used in this example.

# IV. *Image Reconstruction Using Symmetric Convolution*

The theory developed in this chapter applies the symmetric convolution-multiplication property of the discrete trigonometric transforms (DTTs) [34] to the linear image reconstruction problem [10]. The presentation includes a derivation of one and two-dimensional inverse and scalar Wiener filters expressed in the trigonometric transform domain. For finite sequences, point-wise multiplication in the trigonometric transform domain is equivalent to symmetric convolution in the sequence domain. Previous applications of the discrete cosine transform (DCT) to linear image reconstruction [26], [27] provided very good diagonal approximations for certain types of covariance matrices. The DCT cannot, however, simultaneously diagonalize the matrix representing degradation in the linear model. Now, with the symmetric convolution-multiplication property, very good scalar filter approximations are possible because an exact diagonalization of the degradation matrix is achievable, while still retaining the near optimum approximation to the diagonal form of the object covariance matrix. In this chapter, the specific forms needed from among the forty cases of symmetric convolution are first reviewed, and then the derivations of trigonometric one and two-dimensional inverse and scalar Wiener filters are presented. The next section also presents a subtle property on the equivalence of symmetric convolution to linear convolution.

## 4.1 *Equivalence Between Symmetric Convolution and Linear Convolution*

Throughout the remainder of this dissertation, only four of the forty one-dimensional cases of symmetric convolution from Table 3 are considered. This section presents these cases along with the sixteen two-dimensional cases which result from applying the four cases to the rows and

columns of matrices. In the latter two sections of this chapter, these specific cases are used to derive inverse and scalar Wiener filters which recover an object from distorted data. The equivalence between symmetric and linear convolution for appropriately zero-padded sequences [34], [35] is first developed in this section. This equivalence between symmetric and linear convolution is similar to the equivalence which exists between circular and linear convolution which also arises from appropriate zero-padding in the sequence domain [37].

The remaining theory presented in this dissertation requires only four of the forty one-dimensional cases of symmetric convolution from Table 3. These cases, expressed in vector-matrix notation, are

$$\boldsymbol{d}_a = \boldsymbol{S}_{2e,N}^{-1}\left[\boldsymbol{S}_{1e,N}\boldsymbol{h}_a^r \odot \boldsymbol{C}_{2e,N}\theta\right], \tag{64}$$

$$\boldsymbol{d}_a' = -\boldsymbol{C}_{2e,N}^{-1}\left[\boldsymbol{S}_{1e,N}\boldsymbol{h}_a^r \odot \boldsymbol{S}_{2e,N}\theta\right], \tag{65}$$

$$\boldsymbol{d}_s = \boldsymbol{C}_{2e,N}^{-1}\left[\boldsymbol{C}_{1e,N}\boldsymbol{h}_s^r \odot \boldsymbol{C}_{2e,N}\theta\right], \tag{66}$$

and

$$\boldsymbol{d}_s' = \boldsymbol{S}_{2e,N}^{-1}\left[\boldsymbol{C}_{1e,N}\boldsymbol{h}_s^r \odot \boldsymbol{S}_{2e,N}\theta\right]. \tag{67}$$

The vector $\theta$ represents a one-dimensional sequence $\theta(n)$ which is finite and zero outside the interval $0 \le n \le N-1$. It has underlying half-sample symmetry in both the left and right directions (HSHS) in Eqs. (64) and (66) and underlying half-sample antisymmetry in both the left and right directions (HAHA) in Eqs. (65) and (67). Recall that the underlying symmetry for a trigonometric transform is similar to the underlying periodicity required by the discrete Fourier transform (DFT). The vectors $\boldsymbol{h}_a^r$ and $\boldsymbol{h}_s^r$ represent the right halves of impulse responses which are whole-sample antisymmetric and whole-sample symmetric in both the left and right directions (WAWA and WSWS), respectively. These impulse response right halves, $\boldsymbol{h}_a^r$ and $\boldsymbol{h}_s^r$, distort the vector $\theta$ to produce the data sequences $\boldsymbol{d}_a$, $\boldsymbol{d}_a'$, $\boldsymbol{d}_s$, and $\boldsymbol{d}_s'$. The subscripts *'a'* and *'s'* refer to

antisymmetric and symmetric parts and the prime superscript distinguishes between data vectors calculated using different cases of symmetric convolution.

As previously demonstrated, symmetric convolution requires only the right half of a filter's impulse response, because the symmetry is required instead of implied. If the sequence representing the impulse response of the system causing distortion, $h(n)$, is not symmetric initially, it must be decomposed into its antisymmetric and symmetric parts, $h_a^r(n)$ and $h_s^r(n)$, which are represented by the vectors $\boldsymbol{h}_a^r$ and $\boldsymbol{h}_s^r$. Then either Eq. (64) or (65) can convolve the symmetric part and either Eq. (66) or (67) can convolve the antisymmetric part [34], [35]. The four cases in Eqs. (64) - (67) are all based on the type-II DCT for even length sequences, represented by the matrix $\boldsymbol{C}_{2e,N}$. The other matrix transforms in Eqs. (64) - (67) are $\boldsymbol{C}_{1e,N}$, $\boldsymbol{S}_{1e,N}$, and $\boldsymbol{S}_{2e,N}$, which represent the type-I DCT, the type-I discrete sine transform (DST), and the type-II DST, respectively, all for even-length sequences.

A matrix multiplication operation can perform symmetric convolution in the same way that a circulant matrix performs circular convolution. Multiplications by the diagonal matrices $\mathcal{H}_{T,a} = \text{diag}\left\{\boldsymbol{S}_{1e,N}\boldsymbol{h}_a^r\right\}$ and $\mathcal{H}_{T,s} = \text{diag}\left\{\boldsymbol{C}_{1e,N}\boldsymbol{h}_s^r\right\}$ replace the point-wise multiplications in Eqs. (64) - (67). The subscript '$T$' refers to the trigonometric transform domain. These diagonalizations result in the expressions

$$\boldsymbol{d}_a = \boldsymbol{S}_{2e,N}^{-1}\mathcal{H}_{T,a}\boldsymbol{C}_{2e,N}\theta = \boldsymbol{H}_{SC,a}\theta, \tag{68}$$

$$\boldsymbol{d}_a' = -\boldsymbol{C}_{2e,N}^{-1}\mathcal{H}_{T,a}\boldsymbol{S}_{2e,N}\theta = \boldsymbol{H}_{SC,a}'\theta, \tag{69}$$

$$\boldsymbol{d}_s = \boldsymbol{C}_{2e,N}^{-1}\mathcal{H}_{T,s}\boldsymbol{C}_{2e,N}\theta = \boldsymbol{H}_{SC,s}\theta, \tag{70}$$

and

$$\boldsymbol{d}_s' = \boldsymbol{S}_{2e,N}^{-1}\mathcal{H}_{T,s}\boldsymbol{S}_{2e,N}\theta = \boldsymbol{H}_{SC,s}'\theta. \tag{71}$$

The matrices $\boldsymbol{H}_{SC,a}$, $\boldsymbol{H}'_{SC,a}$, $\boldsymbol{H}_{SC,s}$, and $\boldsymbol{H}'_{SC,s}$ are symmetric convolution matrices for the four types of symmetric convolution of interest here. It is important to note that even though $\boldsymbol{H}_{SC,a} \neq \boldsymbol{H}'_{SC,a}$ and $\boldsymbol{H}_{SC,s} \neq \boldsymbol{H}'_{SC,s}$, with appropriate zero-padding in the sequences $\boldsymbol{h}_a^r$, $\boldsymbol{h}_s^r$, and $\theta$ as described in [34] and [35], then $\boldsymbol{d}_a = \boldsymbol{d}'_a$ and $\boldsymbol{d}_s = \boldsymbol{d}'_s$. The equality of this vector representation is assured because of the equality which exists for the sequences $d_a(n) = d'_a(n)$ and $d_s(n) = d'_s(n)$ as demonstrated by Martucci [34]. Appropriate zero-padding assures that the symmetric convolution of the two vectors, $\boldsymbol{h}$ and $\theta$, will equal the result from linear convolution. The equivalence of symmetric and linear convolution is completely analogous to the equivalence of circular and linear convolution which also results from appropriate zero-padding [37]. Thus, provided the vectors $\boldsymbol{h}_a^r$ and $\boldsymbol{h}_s^r$ have the correct underlying symmetry and all sequences are appropriately zero-padded, the results of applying different types of symmetric convolution in Eqs. (68) - (71) are the same because they equal the result from linear convolution.

The equivalence between symmetric and linear convolution applies in two dimensions as well. Consider the lexicographically-ordered vector, $\theta$, representing the $N_1 \times N_2$ object matrix, $\Theta$. Applying a type-II DCT to both dimensions represented within the vector $\theta$ yields

$$\vartheta_{T,ss} = \left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)\theta \tag{72}$$

in the transform domain. The subscript *'ss'* shows that the matrix $\Theta$ has underlying half-sample *symmetry* about both $n_1$ and $n_2$ in the sequence domain (HSHS-HSHS). Equation (72) is the lexicographic equivalent of $\Phi_{T,ss} = \boldsymbol{C}_{2e,N_1} \Theta \boldsymbol{C}_{2e,N_2}^T$ used for the example in the last section of the previous chapter.

Allow the $N_1 \times N_2$ matrix, $\boldsymbol{H}$, which can be viewed as a point spread function (PSF), to represent the other sequence to be convolved. Recall that the matrices $\boldsymbol{H}_{aa}^{rr}$, $\boldsymbol{H}_{as}^{rr}$, $\boldsymbol{H}_{sa}^{rr}$, and $\boldsymbol{H}_{ss}^{rr}$

represent the decomposition of $\boldsymbol{H}$ into its symmetric and antisymmetric parts. The superscript *'rr'* refers to the PSF's right halves about both $n_1$ and $n_2$. If the vectors $\boldsymbol{h}_{aa}^{rr}$, $\boldsymbol{h}_{as}^{rr}$, $\boldsymbol{h}_{sa}^{rr}$, and $\boldsymbol{h}_{ss}^{rr}$ represent lexicographic reorderings of the above decomposed components of $\boldsymbol{H}$, then the correct transformations to apply are

$$\begin{aligned} \boldsymbol{h}_{T,aa}^{rr} &= \left(\boldsymbol{S}_{1e,N_1} \otimes \boldsymbol{S}_{1e,N_2}\right)\boldsymbol{h}_{aa}^{rr}, \qquad \boldsymbol{h}_{T,as}^{rr} = \left(\boldsymbol{C}_{1e,N_1} \otimes \boldsymbol{S}_{1e,N_2}\right)\boldsymbol{h}_{as}^{rr}, \\ \boldsymbol{h}_{T,sa}^{rr} &= \left(\boldsymbol{S}_{1e,N_1} \otimes \boldsymbol{C}_{1e,N_2}\right)\boldsymbol{h}_{sa}^{rr}, \text{ and } \boldsymbol{h}_{T,ss}^{rr} = \left(\boldsymbol{C}_{1e,N_1} \otimes \boldsymbol{C}_{1e,N_2}\right)\boldsymbol{h}_{ss}^{rr}, \end{aligned} \tag{73}$$

based on the different types of symmetry about the $n_1$ and $n_2$ axes. The expressions in Eq. (73) are lexicographically equivalent to the expressions in Eq. (62) because each of the DTTs is separable [27]. Each result, $\boldsymbol{h}_{T,aa}^{rr}$, $\boldsymbol{h}_{T,as}^{rr}$, $\boldsymbol{h}_{T,sa}^{rr}$, and $\boldsymbol{h}_{T,ss}^{rr}$, from Eq. (73) must then be point-multiplied with the vector $\boldsymbol{\vartheta}_{T,ss}$ before inverting to return to the sequence domain. If the lexicographically-ordered vector $\boldsymbol{d}$ represents the final convolved result, then

$$\begin{aligned} \boldsymbol{d} = &\left(\boldsymbol{S}_{2e,N_1}^{-1} \otimes \boldsymbol{S}_{2e,N_2}^{-1}\right)\left[\boldsymbol{h}_{T,aa}^{rr} \odot \boldsymbol{\vartheta}_{T,ss}\right] + \left(\boldsymbol{C}_{2e,N_1}^{-1} \otimes \boldsymbol{S}_{2e,N_2}^{-1}\right)\left[\boldsymbol{h}_{T,as}^{rr} \odot \boldsymbol{\vartheta}_{T,ss}\right] \\ &+ \left(\boldsymbol{S}_{2e,N_1}^{-1} \otimes \boldsymbol{C}_{2e,N_2}^{-1}\right)\left[\boldsymbol{h}_{T,sa}^{rr} \odot \boldsymbol{\vartheta}_{T,ss}\right] + \left(\boldsymbol{C}_{2e,N_1}^{-1} \otimes \boldsymbol{C}_{2e,N_2}^{-1}\right)\left[\boldsymbol{h}_{T,ss}^{rr} \odot \boldsymbol{\vartheta}_{T,ss}\right], \end{aligned} \tag{74}$$

which is lexicographically equivalent to Eq. (63).

Just as in the one-dimensional case, the $N_1N_2 \times N_1N_2$ diagonal matrices

$$\begin{aligned} \boldsymbol{\mathcal{H}}_{T,aa} &= \operatorname{diag}\left\{\boldsymbol{h}_{T,aa}^{rr}\right\}, \qquad \boldsymbol{\mathcal{H}}_{T,as} = \operatorname{diag}\left\{\boldsymbol{h}_{T,as}^{rr}\right\}, \\ \boldsymbol{\mathcal{H}}_{T,sa} &= \operatorname{diag}\left\{\boldsymbol{h}_{T,sa}^{rr}\right\}, \text{ and } \boldsymbol{\mathcal{H}}_{T,ss} = \operatorname{diag}\left\{\boldsymbol{h}_{T,ss}^{rr}\right\}, \end{aligned} \tag{75}$$

can replace the point-wise multiplications in Eq. (74). The $N_1N_2 \times N_1N_2$ symmetric convolution matrices for the two-dimensional case are thus

$$\boldsymbol{H}_{BSC,aa} = \left(\boldsymbol{S}_{2e,N_1}^{-1} \otimes \boldsymbol{S}_{2e,N_2}^{-1}\right)\boldsymbol{\mathcal{H}}_{T,aa}\left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right), \tag{76}$$

$$\boldsymbol{H}_{BSC,as} = \left(\boldsymbol{C}_{2e,N_1}^{-1} \otimes \boldsymbol{S}_{2e,N_2}^{-1}\right)\boldsymbol{\mathcal{H}}_{T,as}\left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right), \tag{77}$$

$$\boldsymbol{H}_{BSC,sa} = \left(\boldsymbol{S}_{2e,N_1}^{-1} \otimes \boldsymbol{C}_{2e,N_2}^{-1}\right) \mathcal{H}_{T,sa} \left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right), \tag{78}$$

and

$$\boldsymbol{H}_{BSC,ss} = \left(\boldsymbol{C}_{2e,N_1}^{-1} \otimes \boldsymbol{C}_{2e,N_2}^{-1}\right) \mathcal{H}_{T,ss} \left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right), \tag{79}$$

where the subscript *'BSC'* indicates the matrices perform *block* symmetric convolution in two dimensions.

Notice that Eqs. (76) - (79) apply the convolution rules from Eqs. (68) and (70), but these convolution rules are not unique since Eq. (69) produces the same result as Eq. (68), and Eq. (71) produces the same result as Eq. (70), as long as appropriate zero-padding exists for all sequences. Just as two representations yielded the same result in each of the two one-dimensional cases, there are now four representations which yield the same result for each of the four two-dimensional cases, producing a total of sixteen cases in all. The transform relation acting on either the rows or the columns in two dimensions may have an equivalent form from Eqs. (68) - (71) substituted for it. Thus the family of two-dimensional symmetric convolution matrices for the antisymmetric-antisymmetric portion of the decomposed PSF is:

$$\boldsymbol{H}_{BSC,aa} = \left(\boldsymbol{S}_{2e,N_1}^{-1} \otimes \boldsymbol{S}_{2e,N_2}^{-1}\right) \mathcal{H}_{T,aa} \left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right), \tag{76}$$

$$\boldsymbol{H}'_{BSC,aa} = -\left(\boldsymbol{C}_{2e,N_1}^{-1} \otimes \boldsymbol{S}_{2e,N_2}^{-1}\right) \mathcal{H}_{T,aa} \left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right), \tag{80}$$

$$\boldsymbol{H}''_{BSC,aa} = -\left(\boldsymbol{S}_{2e,N_1}^{-1} \otimes \boldsymbol{C}_{2e,N_2}^{-1}\right) \mathcal{H}_{T,aa} \left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right), \tag{81}$$

and

$$\boldsymbol{H}'''_{BSC,aa} = \left(\boldsymbol{C}_{2e,N_1}^{-1} \otimes \boldsymbol{C}_{2e,N_2}^{-1}\right) \mathcal{H}_{T,aa} \left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right). \tag{82}$$

The family for the antisymmetric-symmetric portion is:

$$\boldsymbol{H}_{BSC,as} = \left(\boldsymbol{C}_{2e,N_1}^{-1} \otimes \boldsymbol{S}_{2e,N_2}^{-1}\right) \mathcal{H}_{T,as} \left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right), \tag{77}$$

$$\boldsymbol{H}'_{BSC,as} = \left(\boldsymbol{S}_{2e,N_1}^{-1} \otimes \boldsymbol{S}_{2e,N_2}^{-1}\right) \mathcal{H}_{T,as} \left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right), \tag{83}$$

$$\boldsymbol{H}''_{BSC,as} = -\left(\boldsymbol{C}_{2e,N_1}^{-1} \otimes \boldsymbol{C}_{2e,N_2}^{-1}\right) \mathcal{H}_{T,as} \left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right), \tag{84}$$

and $$\boldsymbol{H}'''_{BSC,as} = -\left(\boldsymbol{S}^{-1}_{2e,N_1} \otimes \boldsymbol{C}^{-1}_{2e,N_2}\right)\mathcal{H}_{T,as}\left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right). \quad (85)$$

The family for the symmetric-antisymmetric portion is:

$$\boldsymbol{H}_{BSC,sa} = \left(\boldsymbol{S}^{-1}_{2e,N_1} \otimes \boldsymbol{C}^{-1}_{2e,N_2}\right)\mathcal{H}_{T,sa}\left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right), \quad (78)$$

$$\boldsymbol{H}'_{BSC,sa} = -\left(\boldsymbol{C}^{-1}_{2e,N_1} \otimes \boldsymbol{C}^{-1}_{2e,N_2}\right)\mathcal{H}_{T,sa}\left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right), \quad (86)$$

$$\boldsymbol{H}''_{BSC,sa} = \left(\boldsymbol{S}^{-1}_{2e,N_1} \otimes \boldsymbol{S}^{-1}_{2e,N_2}\right)\mathcal{H}_{T,sa}\left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right), \quad (87)$$

and $$\boldsymbol{H}'''_{BSC,sa} = -\left(\boldsymbol{C}^{-1}_{2e,N_1} \otimes \boldsymbol{S}^{-1}_{2e,N_2}\right)\mathcal{H}_{T,sa}\left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right). \quad (88)$$

The family for the symmetric-symmetric portion is:

$$\boldsymbol{H}_{BSC,ss} = \left(\boldsymbol{C}^{-1}_{2e,N_1} \otimes \boldsymbol{C}^{-1}_{2e,N_2}\right)\mathcal{H}_{T,ss}\left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right), \quad (79)$$

$$\boldsymbol{H}'_{BSC,ss} = \left(\boldsymbol{S}^{-1}_{2e,N_1} \otimes \boldsymbol{C}^{-1}_{2e,N_2}\right)\mathcal{H}_{T,ss}\left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right), \quad (89)$$

$$\boldsymbol{H}''_{BSC,ss} = \left(\boldsymbol{C}^{-1}_{2e,N_1} \otimes \boldsymbol{S}^{-1}_{2e,N_2}\right)\mathcal{H}_{T,ss}\left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right), \quad (90)$$

and $$\boldsymbol{H}'''_{BSC,ss} = \left(\boldsymbol{S}^{-1}_{2e,N_1} \otimes \boldsymbol{S}^{-1}_{2e,N_2}\right)\mathcal{H}_{T,ss}\left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right). \quad (91)$$

Here again none of the symmetric convolution matrices within a particular family are exactly equal, *i.e.* $\boldsymbol{H}_{BSC,aa} \neq \boldsymbol{H}'_{BSC,aa} \neq \boldsymbol{H}''_{BSC,aa} \neq \boldsymbol{H}'''_{BSC,aa}$, $\boldsymbol{H}_{BSC,as} \neq \boldsymbol{H}'_{BSC,as} \neq \boldsymbol{H}''_{BSC,as} \neq \boldsymbol{H}'''_{BSC,as}$, $\boldsymbol{H}_{BSC,sa} \neq \boldsymbol{H}'_{BSC,sa} \neq \boldsymbol{H}''_{BSC,sa} \neq \boldsymbol{H}'''_{BSC,sa}$, and $\boldsymbol{H}_{BSC,ss} \neq \boldsymbol{H}'_{BSC,ss} \neq \boldsymbol{H}''_{BSC,ss} \neq \boldsymbol{H}'''_{BSC,ss}$. However, with appropriate zero-padding, the vectors $\boldsymbol{d}_{aa} = \boldsymbol{H}_{BSC,aa}\theta$, $\boldsymbol{d}'_{aa} = \boldsymbol{H}'_{BSC,aa}\theta$, $\boldsymbol{d}''_{aa} = \boldsymbol{H}''_{BSC,aa}\theta$, and $\boldsymbol{d}'''_{aa} = \boldsymbol{H}'''_{BSC,aa}\theta$ will all be equal; the vectors $\boldsymbol{d}_{as} = \boldsymbol{H}_{BSC,as}\theta$, $\boldsymbol{d}'_{as} = \boldsymbol{H}'_{BSC,as}\theta$, $\boldsymbol{d}''_{as} = \boldsymbol{H}''_{BSC,as}\theta$, and $\boldsymbol{d}'''_{as} = \boldsymbol{H}'''_{BSC,as}\theta$ will all be equal; the vectors $\boldsymbol{d}_{sa} = \boldsymbol{H}_{BSC,sa}\theta$, $\boldsymbol{d}'_{sa} = \boldsymbol{H}'_{BSC,sa}\theta$, $\boldsymbol{d}''_{sa} = \boldsymbol{H}''_{BSC,sa}\theta$, and $\boldsymbol{d}'''_{sa} = \boldsymbol{H}'''_{BSC,sa}\theta$ will all be equal; and the vectors $\boldsymbol{d}_{ss} = \boldsymbol{H}_{BSC,ss}\theta$, $\boldsymbol{d}'_{ss} = \boldsymbol{H}'_{BSC,ss}\theta$, $\boldsymbol{d}''_{ss} = \boldsymbol{H}''_{BSC,ss}\theta$, and $\boldsymbol{d}'''_{ss} = \boldsymbol{H}'''_{BSC,ss}\theta$ will all be equal. The resulting sequences will also equal the sequence which arises from circular convolution with appropriate zero padding, because all are equal to the result

from linear convolution. The same vector $\boldsymbol{d}$ will result if any of the different equivalent forms in Eqs. (76) - (91) are substituted into Eq. (74), as long the vectors $\boldsymbol{h}$ and $\theta$ are appropriately zero padded. All of the different forms of symmetric convolution in Eqs. (76) - (91) use the vector-matrix form [9] of Martucci's [34] symmetric convolution-multiplication property for trigonometric transforms derived in the previous chapter.

The equivalence of symmetric convolution to linear convolution as outlined in this section holds in the trigonometric transform domain for any linear shift-invariant filtering application. These results are applied in the following sections by recasting some traditional linear image reconstruction filtering operations into the trigonometric transform domain.

### 4.2 *Inverse Filtering in the Trigonometric Transform Domain*

The results of the previous sections are valid for any finite sequences represented by the vectors $\boldsymbol{d}$, $\boldsymbol{h}$, and $\theta$, which are all lexicographically-ordered for the two-dimensional case. The only underlying assumption was appropriate zero-padding to demonstrate the equivalence of the convolutional forms. In this section the vector $\boldsymbol{d}$ represents a detected image which arises from blurring an original object vector $\theta$ by a point spread function (PSF) represented lexicographically as $\boldsymbol{h}$. The goal of the inverse filters derived in this section is to recover a vector estimate of the object, $\hat{\theta}$, from the vector $\boldsymbol{d}$ in the trigonometric transform domain. This classical problem as it applies to image reconstruction finds the two-dimensional impulse response of a filter, represented lexicographically by the vector $\boldsymbol{f}$, which recovers $\hat{\theta}$ from $\boldsymbol{d}$, given knowledge of $\boldsymbol{h}$. This is the same problem which generated Fourier transform domain solutions presented in Section 2.2 as background. The following subsections present equivalent representations in the trigonometric transform domain, first for one and then for two dimensions.

**4.2.1** ***The One-Dimensional Inverse Filter for Trigonometric Transforms.*** The development of this subsection derives a result in the trigonometric transform domain similar to Eq. (10) which presented the one-dimensional inverse filter in the Fourier domain. Using the same notation as before whereby $\boldsymbol{d} = \boldsymbol{H}\theta$ and $\hat{\theta} = \boldsymbol{F}\boldsymbol{d} = \boldsymbol{F}\boldsymbol{H}\theta$, understand that in this case the matrices $\boldsymbol{H}$ and $\boldsymbol{F}$ represent one-dimensional *symmetric convolution* matrices rather than circular convolution matrices. The notation here drops the subscript *'SC'* which was useful in previous sections to distinguish between symmetric and circular or skew-circular convolution. The discussion from here forward concerns itself exclusively with symmetric convolution and trigonometric transforms. The *'BSC'* subscripts will likewise not appear in two-dimensional block symmetric convolution matrices, nor will the subscript *'T'* appear in trigonometric transform domain quantities.

The symmetric convolution matrices $\boldsymbol{H}$ and $\boldsymbol{F}$ may, in general, be asymmetric, which requires a decomposition into their symmetric and antisymmetric parts so that $\boldsymbol{d} = (\boldsymbol{H}_a + \boldsymbol{H}_s)\theta$ and

$$\hat{\theta} = (\boldsymbol{F}_a + \boldsymbol{F}_s)\boldsymbol{d} = (\boldsymbol{F}_a + \boldsymbol{F}_s)(\boldsymbol{H}_a + \boldsymbol{H}_s)\theta. \tag{92}$$

Expanding Eq. (92) and making substitutions from Eqs. (68) - (71) results in

$$\begin{aligned}\hat{\theta} = \Big[&\left(-\boldsymbol{C}_{2e,N}^{-1}\mathcal{J}_a\boldsymbol{S}_{2e,N}\right)\left(\boldsymbol{S}_{2e,N}^{-1}\mathcal{H}_a\boldsymbol{C}_{2e,N}\right) + \left(-\boldsymbol{C}_{2e,N}^{-1}\mathcal{J}_a\boldsymbol{S}_{2e,N}\right)\left(\boldsymbol{S}_{2e,N}^{-1}\mathcal{H}_s\boldsymbol{S}_{2e,N}\right) \\ &+ \left(\boldsymbol{C}_{2e,N}^{-1}\mathcal{J}_s\boldsymbol{C}_{2e,N}\right)\left(-\boldsymbol{C}_{2e,N}^{-1}\mathcal{H}_a\boldsymbol{S}_{2e,N}\right) + \left(\boldsymbol{C}_{2e,N}^{-1}\mathcal{J}_s\boldsymbol{C}_{2e,N}\right)\left(\boldsymbol{C}_{2e,N}^{-1}\mathcal{H}_s\boldsymbol{C}_{2e,N}\right)\Big]\theta.\end{aligned} \tag{93}$$

Equation (93) incorporates substitutions of the different equivalent forms of symmetric convolution based on the assumption that the sequence domain data vector, $\boldsymbol{d}$, is the result of linear convolution. This assumption implies that sufficient zero-padding [34], [35] exists in the vectors $\boldsymbol{h}$ and $\theta$ to ensure their equivalence. The innermost matrices inside each term of Eq. (93) multiply to identity, so that exact reconstruction to yield the vector $\hat{\theta} = \theta$, requires

$$\mathcal{J}_s\mathcal{H}_s - \mathcal{J}_a\mathcal{H}_a = \boldsymbol{I}, \tag{94}$$

and

$$\mathcal{F}_a\mathcal{H}_s + \mathcal{F}_s\mathcal{H}_a = \mathbf{0}, \tag{95}$$

where $\boldsymbol{I}$ is an $N \times N$ identity matrix, and $\mathbf{0}$ is an $N \times N$ zero matrix. Because all the matrices in Eqs. (94) and (95) are diagonal, the equations are equivalent to the matrix equation

$$\begin{bmatrix} -\mathcal{H}_a(k) & \mathcal{H}_s(k) \\ \mathcal{H}_s(k) & \mathcal{H}_a(k) \end{bmatrix} \begin{bmatrix} \mathcal{F}_a(k) \\ \mathcal{F}_s(k) \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \tag{96}$$

for $k = 0, 1, \ldots, N-1$. The terms $\mathcal{H}_a(k)$, $\mathcal{H}_s(k)$, $\mathcal{F}_a(k)$, and $\mathcal{F}_s(k)$ in Eq. (96) represent the $k$-$k$th terms, $[\mathcal{H}_a]_{kk}$, $[\mathcal{H}_s]_{kk}$, $[\mathcal{F}_a]_{kk}$, and $[\mathcal{F}_s]_{kk}$, of the matrices $\mathcal{H}_a$, $\mathcal{H}_s$, $\mathcal{F}_a$, and $\mathcal{F}_s$, respectively. Solving Eq. (96) for $\mathcal{F}_a(k)$ and $\mathcal{F}_s(k)$ yields

$$\mathcal{F}_a(k) = \frac{-\mathcal{H}_a(k)}{\mathcal{H}_a^2(k) + \mathcal{H}_s^2(k)} \tag{97}$$

and

$$\mathcal{F}_s(k) = \frac{\mathcal{H}_s(k)}{\mathcal{H}_a^2(k) + \mathcal{H}_s^2(k)}. \tag{98}$$

Equations (97) and (98) are the one-dimensional inverse filter expressed in the trigonometric transform domain for symmetrically convolved sequences. If the sequence $h(n)$ possesses strictly whole-sample symmetry, *i.e.*, $h_a(n) = \mathcal{H}_a(k) = 0$, then the inverse filter reduces to

$$\mathcal{F}_s(k) = \frac{1}{\mathcal{H}_s(k)}. \tag{99}$$

Equation (99) is similar to the Fourier domain result in Eq. (10). The following subsection shows similar results for two dimensions.

**4.2.2** ***The Two-Dimensional Inverse Filter for Trigonometric Transforms.*** In two dimensions, $\boldsymbol{H}$ and $\boldsymbol{F}$ become two-dimensional block symmetric convolution matrices which may also in general be asymmetric. They therefore require a decomposition into their symmetric and antisymmetric parts about both the $n_1$ and $n_2$ axes so that $\boldsymbol{d} = (\boldsymbol{H}_{aa} + \boldsymbol{H}_{as} + \boldsymbol{H}_{sa} + \boldsymbol{H}_{ss})\theta$ and

$$\begin{aligned}\hat{\theta} &= \left(\boldsymbol{F}_{aa} + \boldsymbol{F}_{as} + \boldsymbol{F}_{sa} + \boldsymbol{F}_{ss}\right)\boldsymbol{d} \\ &= \left(\boldsymbol{F}_{aa} + \boldsymbol{F}_{as} + \boldsymbol{F}_{sa} + \boldsymbol{F}_{ss}\right)\left(\boldsymbol{H}_{aa} + \boldsymbol{H}_{as} + \boldsymbol{H}_{sa} + \boldsymbol{H}_{ss}\right)\theta.\end{aligned} \tag{100}$$

Expanding Eq. (100) and making substitutions from the equivalent forms in Eqs. (76) - (91) results in

$$\begin{aligned}\hat{\theta} = \left(\boldsymbol{C}_{2e,N_1}^{-1} \otimes \boldsymbol{C}_{2e,N_2}^{-1}\right)\Big\{ & \mathcal{F}_{aa}\left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right)\Big[\left(\boldsymbol{S}_{2e,N_1}^{-1} \otimes \boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right) \\ & + \left(\boldsymbol{S}_{2e,N_1}^{-1} \otimes \boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right) \\ & + \left(\boldsymbol{S}_{2e,N_1}^{-1} \otimes \boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right) \\ & + \left(\boldsymbol{S}_{2e,N_1}^{-1} \otimes \boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right)\Big] \\ & + \mathcal{F}_{as}\left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right)\Big[\left(\boldsymbol{C}_{2e,N_1}^{-1} \otimes \boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right) \\ & - \left(\boldsymbol{C}_{2e,N_1}^{-1} \otimes \boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right) \\ & + \left(\boldsymbol{C}_{2e,N_1}^{-1} \otimes \boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right) \\ & - \left(\boldsymbol{C}_{2e,N_1}^{-1} \otimes \boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right)\Big] \\ & + \mathcal{F}_{sa}\left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)\Big[\left(\boldsymbol{S}_{2e,N_1}^{-1} \otimes \boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right) \\ & + \left(\boldsymbol{S}_{2e,N_1}^{-1} \otimes \boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right) \\ & - \left(\boldsymbol{S}_{2e,N_1}^{-1} \otimes \boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right) \\ & - \left(\boldsymbol{S}_{2e,N_1}^{-1} \otimes \boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)\Big] \\ & + \mathcal{F}_{ss}\left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)\Big[\left(\boldsymbol{C}_{2e,N_1}^{-1} \otimes \boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right) \\ & - \left(\boldsymbol{C}_{2e,N_1}^{-1} \otimes \boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right) \\ & - \left(\boldsymbol{C}_{2e,N_1}^{-1} \otimes \boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right) \\ & + \left(\boldsymbol{C}_{2e,N_1}^{-1} \otimes \boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)\Big]\Big\}\theta.\end{aligned} \tag{101}$$

Equation (101) incorporates substitutions of the different equivalent forms of the convolution matrices in Eqs. (76) - (91). The substitutions are permissible because of the assumption that the lexicographically-ordered data vector, $\boldsymbol{d}$, arises as the result of linear convolution in the sequence domain.

For this two-dimensional case, the interior matrices in each term of Eq. (101) again multiply to identity, so that exact reconstruction requires

$$\mathcal{F}_{aa}\mathcal{H}_{aa} - \mathcal{F}_{as}\mathcal{H}_{as} - \mathcal{F}_{sa}\mathcal{H}_{sa} + \mathcal{F}_{ss}\mathcal{H}_{ss} = \boldsymbol{I}, \tag{102}$$

$$\mathcal{F}_{aa}\mathcal{H}_{as} + \mathcal{F}_{as}\mathcal{H}_{aa} - \mathcal{F}_{sa}\mathcal{H}_{ss} - \mathcal{F}_{ss}\mathcal{H}_{sa} = \boldsymbol{0}, \tag{103}$$

$$\mathcal{F}_{aa}\mathcal{H}_{sa} - \mathcal{F}_{as}\mathcal{H}_{ss} + \mathcal{F}_{sa}\mathcal{H}_{aa} - \mathcal{F}_{ss}\mathcal{H}_{as} = \boldsymbol{0}, \tag{104}$$

and

$$\mathcal{F}_{aa}\mathcal{H}_{ss} + \mathcal{F}_{as}\mathcal{H}_{sa} + \mathcal{F}_{sa}\mathcal{H}_{as} + \mathcal{F}_{ss}\mathcal{H}_{aa} = \boldsymbol{0}. \tag{105}$$

Each matrix in Eqs. (102) - (105) is diagonal, which generates the matrix equation

$$\begin{bmatrix} \mathcal{H}_{aa}(k_1,k_2) & -\mathcal{H}_{as}(k_1,k_2) & -\mathcal{H}_{sa}(k_1,k_2) & \mathcal{H}_{ss}(k_1,k_2) \\ \mathcal{H}_{as}(k_1,k_2) & \mathcal{H}_{aa}(k_1,k_2) & -\mathcal{H}_{ss}(k_1,k_2) & -\mathcal{H}_{sa}(k_1,k_2) \\ \mathcal{H}_{sa}(k_1,k_2) & -\mathcal{H}_{ss}(k_1,k_2) & \mathcal{H}_{aa}(k_1,k_2) & -\mathcal{H}_{as}(k_1,k_2) \\ \mathcal{H}_{ss}(k_1,k_2) & \mathcal{H}_{sa}(k_1,k_2) & \mathcal{H}_{as}(k_1,k_2) & \mathcal{H}_{aa}(k_1,k_2) \end{bmatrix} \begin{bmatrix} \mathcal{F}_{aa}(k_1,k_2) \\ \mathcal{F}_{as}(k_1,k_2) \\ \mathcal{F}_{sa}(k_1,k_2) \\ \mathcal{F}_{ss}(k_1,k_2) \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \tag{106}$$

for $k_1 = 0,\ 1,\ \ldots,\ N_1 - 1$ and $k_2 = 0,\ 1,\ \ldots,\ N_2 - 1$. The terms $\mathcal{H}_{aa}(k_1,k_2)$, $\mathcal{H}_{as}(k_1,k_2)$, $\mathcal{H}_{sa}(k_1,k_2)$, $\mathcal{H}_{ss}(k_1,k_2)$, $\mathcal{F}_{aa}(k_1,k_2)$, $\mathcal{F}_{as}(k_1,k_2)$, $\mathcal{F}_{sa}(k_1,k_2)$, and $\mathcal{F}_{ss}(k_1,k_2)$ in Eq. (106) represent the diagonal elements of the $N_1N_2 \times N_1N_2$ matrices $\mathcal{H}_{aa}$, $\mathcal{H}_{as}$, $\mathcal{H}_{sa}$, $\mathcal{H}_{ss}$, $\mathcal{F}_{aa}$, $\mathcal{F}_{as}$, $\mathcal{F}_{sa}$, and $\mathcal{F}_{ss}$, respectively. Equation (106) has the solution

$$\mathcal{F}_{aa}(k_1,k_2) = \frac{1}{\mathcal{A}(k_1,k_2)} \left\{ \begin{array}{l} \mathcal{H}_{aa}^3(k_1,k_2) + \mathcal{H}_{aa}(k_1,k_2)\mathcal{H}_{as}^2(k_1,k_2) + \mathcal{H}_{aa}(k_1,k_2)\mathcal{H}_{sa}^2(k_1,k_2) \\ \quad - \mathcal{H}_{aa}(k_1,k_2)\mathcal{H}_{ss}^2(k_1,k_2) + 2\mathcal{H}_{as}(k_1,k_2)\mathcal{H}_{sa}(k_1,k_2)\mathcal{H}_{ss}(k_1,k_2) \end{array} \right\}, \tag{107}$$

$$\mathcal{F}_{as}(k_1,k_2) = \frac{-1}{\mathcal{A}(k_1,k_2)} \left\{ \begin{array}{l} \mathcal{H}_{as}^3(k_1,k_2) + \mathcal{H}_{as}(k_1,k_2)\mathcal{H}_{aa}^2(k_1,k_2) - \mathcal{H}_{as}(k_1,k_2)\mathcal{H}_{sa}^2(k_1,k_2) \\ \quad + \mathcal{H}_{as}(k_1,k_2)\mathcal{H}_{ss}^2(k_1,k_2) + 2\mathcal{H}_{aa}(k_1,k_2)\mathcal{H}_{sa}(k_1,k_2)\mathcal{H}_{ss}(k_1,k_2) \end{array} \right\}, \tag{108}$$

$$\mathcal{F}_{sa}(k_1,k_2) = \frac{-1}{\mathcal{A}(k_1,k_2)} \left\{ \begin{array}{l} \mathcal{H}_{sa}^3(k_1,k_2) + \mathcal{H}_{sa}(k_1,k_2)\mathcal{H}_{aa}^2(k_1,k_2) - \mathcal{H}_{sa}(k_1,k_2)\mathcal{H}_{as}^2(k_1,k_2) \\ \quad + \mathcal{H}_{sa}(k_1,k_2)\mathcal{H}_{ss}^2(k_1,k_2) + 2\mathcal{H}_{aa}(k_1,k_2)\mathcal{H}_{as}(k_1,k_2)\mathcal{H}_{ss}(k_1,k_2) \end{array} \right\}, \tag{109}$$

and

$$\mathcal{F}_{ss}(k_1,k_2) = \frac{1}{\mathcal{A}(k_1,k_2)} \left\{ \begin{array}{l} \mathcal{H}_{ss}^3(k_1,k_2) - \mathcal{H}_{ss}(k_1,k_2)\mathcal{H}_{aa}^2(k_1,k_2) + \mathcal{H}_{ss}(k_1,k_2)\mathcal{H}_{as}^2(k_1,k_2) \\ \quad + \mathcal{H}_{ss}(k_1,k_2)\mathcal{H}_{sa}^2(k_1,k_2) + 2\mathcal{H}_{aa}(k_1,k_2)\mathcal{H}_{as}(k_1,k_2)\mathcal{H}_{sa}(k_1,k_2) \end{array} \right\}, \tag{110}$$

where the determinant of the $4 \times 4$ matrix in Eq. (106) is given by

$$\mathcal{A}(k_1,k_2) = \left\{\left[\mathcal{H}_{aa}(k_1,k_2) + \mathcal{H}_{ss}(k_1,k_2)\right]^2 + \left[\mathcal{H}_{as}(k_1,k_2) - \mathcal{H}_{sa}(k_1,k_2)\right]^2\right\} \times \left\{\left[\mathcal{H}_{aa}(k_1,k_2) - \mathcal{H}_{ss}(k_1,k_2)\right]^2 + \left[\mathcal{H}_{as}(k_1,k_2) + \mathcal{H}_{sa}(k_1,k_2)\right]^2\right\}. \tag{111}$$

Equations (107) - (111) are the two-dimensional inverse filter for trigonometric transforms. If the sequence $h(n_1,n_2)$ possesses strictly whole-sample symmetry about the $n_1$ and $n_2$ axes, *i.e.*, $h_{aa}(n_1,n_2) = \mathcal{H}_{aa}(k_1,k_2) = 0$, $h_{as}(n_1,n_2) = \mathcal{H}_{as}(k_1,k_2) = 0$, and $h_{sa}(n_1,n_2) = \mathcal{H}_{sa}(k_1,k_2) = 0$, then the two-dimensional inverse filter reduces to

$$\mathcal{F}_{ss}(k_1,k_2) = \frac{1}{\mathcal{H}_{ss}(k_1,k_2)}. \tag{112}$$

Equation (112) closely resembles its Fourier domain equivalent in Eq. (12).

The trigonometric transform domain realizations of the inverse filter for symmetrically-convolved one and two-dimensional sequences expressed in Eqs. (97), (98), and (107) - (111) suffer from the same high-frequency gain problem which plagued their Fourier domain equivalent forms for noisy sequences. The problem becomes quite serious when the image model expands to incorporate noise with a uniform power spectral density across all frequencies. In the next section, a method is presented of regularizing the high frequency gain of the inverse filter in the presence of noise.

### 4.3 *Wiener Filtering in the Trigonometric Transform Domain*

In this section, a derivation of the scalar Wiener filter is presented in the trigonometric transform domain [12]. A Wiener filter introduces a degree of regularization to the inverse problem and is more capable of filtering data models with noise, as explained in the background section on Wiener filtering in Chapter II. To incorporate noise into the data model of the previous section, the model must have an added term so that it now becomes $\boldsymbol{d} = \boldsymbol{H}\theta + \boldsymbol{w}$. The matrix $\boldsymbol{H}$ is a symmetric convolution matrix for the one-dimensional case and a block symmetric convo-

lution matrix for the two-dimensional case. The vector $\boldsymbol{w}$ is again a zero-mean uniform-variance noise vector whose samples are uncorrelated both with the samples of the object vector, $\theta$, and with each other. The vectors $\boldsymbol{d}$ and $\theta$ represent finite sequences for one dimension and are lexicographically-ordered representations of finite matrices for two dimensions. The object vector, $\theta$, is itself random with constant mean vector $\mu_\theta = \overline{\theta}$, which can be assumed to equal $\mathbf{0}$ without loss of generality [28], and correlation matrix $\boldsymbol{R}_{\theta\theta} = \overline{\theta\theta^T}$. The Wiener filter seeks the vector estimate, $\hat{\theta}$, of $\theta$ which minimizes the mean squared error, $\overline{\varepsilon^T\varepsilon}$, where the vector $\varepsilon = \theta - \hat{\theta}$.

The solution to this problem from [28] appeared previously for circulant and block circulant matrices in the Fourier transform case. The equivalent to Eq. (14) for symmetric convolution is

$$\hat{\theta} = \boldsymbol{R}_{\theta\theta}\boldsymbol{H}^T\left[\boldsymbol{H}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}^T + \boldsymbol{R}_{ww}\right]^{-1}\boldsymbol{d}, \tag{113}$$

which produces the recovery filter

$$\boldsymbol{F} = \boldsymbol{R}_{\theta\theta}\boldsymbol{H}^T\left[\boldsymbol{H}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}^T + \boldsymbol{R}_{ww}\right]^{-1}. \tag{114}$$

Recall that the background section on Wiener filtering in Chapter II described vector and scalar Wiener filters in one and two dimensions expressed in the Fourier transform domain. In that case the matrices $\boldsymbol{F}$ and $\boldsymbol{H}$ were circulant for one-dimensional filters and block circulant for two dimensional filters. Here in the trigonometric case, the matrices $\boldsymbol{F}$ and $\boldsymbol{H}$ will represent symmetric convolution matrices for one-dimensional filters and block symmetric convolution matrices for two-dimensional filters.

The Fourier-domain scalar Wiener filters presented for background in Eqs. (23) and (27) for one and two dimensions under the assumption of wide-sense stationarity for the object, provide a good approximation to the vector or generalized Wiener filter [38]. The approximation

arose because the discrete Fourier transform did not exactly diagonalize the symmetric Toeplitz form of the correlation matrix for a wide-sense stationary object [48]. Note that the noise is already stationary because it has zero mean and its samples are uncorrelated with each other.

In the derivations which follow in the trigonometric transform domain, the data vector, $\boldsymbol{d}$, must arise as the result of a linear convolution of the object vector, $\theta$, with the point spread function, $\boldsymbol{h}$. This assumption of linear convolution allows the substitution of different equivalent forms of the two-dimensional convolution matrices from Eqs. (76) - (91) into the components which result from decomposing the matrix $\boldsymbol{H}$ in Eq. (114). In the one-dimensional problem, the one-dimensional convolution matrices from Eqs. (68) - (71) may be substituted. This assumption of linear convolution also exists in Fourier-domain derivations because the processes which generate blurred data are linear and not circular in nature. Circular convolution arises because it is the underlying form of convolution for DFTs and it is convenient mathematically to process the data using DFTs. However, for the data to arise from linear convolution, the object must be zero-padded, but a zero-padded or support-constrained sequence cannot be wide-sense stationary. The assumption of linear convolution underlying the process which generated the blurred data thus seems to contradict the assumption of wide-sense stationarity for the object.

Hunt and Cannon [25] addressed this conundrum, and their work is further refined in [51]. Their work chooses the more accurate nonstationary object model with support constraints to account for the zeros which must exist to be equivalent to linear convolution. A model without support constraints does not reflect the true statistics of the object. The correlation matrix of an object with support constraints has zeros which appear due to the zero-padding which must exist for linear convolution to appear. These zeros do not appear for an object without support constraints. A model without support constraints is, however, more mathematically tractable because it is well-approximated by a diagonal matrix in the transform domain [48]. In a support-

constrained model, a higher degree of correlation exists between frequencies in the Fourier domain [36]. The purpose of this research is to find scalar filters which require good diagonal approximations in the trigonometric transform domain. In this case it is better not to use support constraints in the model so that good diagonal approximations result. For vector filters, the support-constrained nonstationary object model is better because it exhibits enhanced performance by incorporating the off-diagonal correlations into the filter design [14]. The choice of a model without support constraints is also more appropriate in this scalar filter derivation because this effort is the first attempt to apply the symmetric convolution-multiplication property to a Wiener filter. The original Fourier-domain Wiener filters were all unconstrained [15], [27], [28], [43], [48], so this choice provides a better comparison to earlier work. In the following discussion, the one-dimensional scalar Wiener filter for trigonometric transforms is derived first, and then its two-dimensional equivalent is presented.

**4.3.1** ***The One-Dimensional Scalar Wiener Filter for Trigonometric Transforms.*** The first step in deriving the one-dimensional scalar Wiener filter is to substitute the decomposed symmetric convolution matrices $\boldsymbol{F} = \boldsymbol{F}_a + \boldsymbol{F}_s$ and $\boldsymbol{H} = \boldsymbol{H}_a + \boldsymbol{H}_s$ into Eq. (114). Making these substitutions and then bringing the bracketed expression over to the left side produces

$$\left(\boldsymbol{F}_a + \boldsymbol{F}_s\right)\left[\left(\boldsymbol{H}_a + \boldsymbol{H}_s\right)\boldsymbol{R}_{\theta\theta}\left(\boldsymbol{H}_a + \boldsymbol{H}_s\right)^T + \boldsymbol{R}_{ww}\right] = \boldsymbol{R}_{\theta\theta}\left(\boldsymbol{H}_a + \boldsymbol{H}_s\right)^T. \tag{115}$$

This derivation then follows a similar procedure as the inverse filter derivation. It expands Eq. (115) and then substitutes the equivalent forms from Eqs. (68) - (71) to produce

$$\begin{aligned}
&\left(-\boldsymbol{C}_{2e,N}^{-1}\mathcal{F}_a\boldsymbol{S}_{2e,N}\right)\left(\boldsymbol{S}_{2e,N}^{-1}\mathcal{H}_a\boldsymbol{C}_{2e,N}\right)\boldsymbol{R}_{\theta\theta}\left(\boldsymbol{S}_{2e,N}^{-1}\mathcal{H}_a\boldsymbol{C}_{2e,N}\right)^T \\
&+\left(-\boldsymbol{C}_{2e,N}^{-1}\mathcal{F}_a\boldsymbol{S}_{2e,N}\right)\left(\boldsymbol{S}_{2e,N}^{-1}\mathcal{H}_a\boldsymbol{C}_{2e,N}\right)\boldsymbol{R}_{\theta\theta}\left(\boldsymbol{C}_{2e,N}^{-1}\mathcal{H}_s\boldsymbol{C}_{2e,N}\right)^T \\
&+\left(\boldsymbol{S}_{2e,N}^{-1}\mathcal{F}_a\boldsymbol{C}_{2e,N}\right)\left(\boldsymbol{C}_{2e,N}^{-1}\mathcal{H}_s\boldsymbol{C}_{2e,N}\right)\boldsymbol{R}_{\theta\theta}\left(\boldsymbol{S}_{2e,N}^{-1}\mathcal{H}_a\boldsymbol{C}_{2e,N}\right)^T \\
&+\left(\boldsymbol{S}_{2e,N}^{-1}\mathcal{F}_a\boldsymbol{C}_{2e,N}\right)\left(\boldsymbol{C}_{2e,N}^{-1}\mathcal{H}_s\boldsymbol{C}_{2e,N}\right)\boldsymbol{R}_{\theta\theta}\left(\boldsymbol{C}_{2e,N}^{-1}\mathcal{H}_s\boldsymbol{C}_{2e,N}\right)^T \\
&+\left(\boldsymbol{S}_{2e,N}^{-1}\mathcal{F}_s\boldsymbol{S}_{2e,N}\right)\left(\boldsymbol{S}_{2e,N}^{-1}\mathcal{H}_a\boldsymbol{C}_{2e,N}\right)\boldsymbol{R}_{\theta\theta}\left(\boldsymbol{S}_{2e,N}^{-1}\mathcal{H}_a\boldsymbol{C}_{2e,N}\right)^T\cdots
\end{aligned}$$

$$
\begin{aligned}
&+\left(\boldsymbol{S}_{2e,N}^{-1}\boldsymbol{\mathcal{F}}_s\boldsymbol{S}_{2e,N}\right)\left(\boldsymbol{S}_{2e,N}^{-1}\boldsymbol{\mathcal{H}}_a\boldsymbol{C}_{2e,N}\right)\boldsymbol{R}_{\theta\theta}\left(\boldsymbol{C}_{2e,N}^{-1}\boldsymbol{\mathcal{H}}_s\boldsymbol{C}_{2e,N}\right)^T\\
&+\left(\boldsymbol{C}_{2e,N}^{-1}\boldsymbol{\mathcal{F}}_s\boldsymbol{C}_{2e,N}\right)\left(\boldsymbol{C}_{2e,N}^{-1}\boldsymbol{\mathcal{H}}_s\boldsymbol{C}_{2e,N}\right)\boldsymbol{R}_{\theta\theta}\left(\boldsymbol{S}_{2e,N}^{-1}\boldsymbol{\mathcal{H}}_a\boldsymbol{C}_{2e,N}\right)^T\\
&+\left(\boldsymbol{C}_{2e,N}^{-1}\boldsymbol{\mathcal{F}}_s\boldsymbol{C}_{2e,N}\right)\left(\boldsymbol{C}_{2e,N}^{-1}\boldsymbol{\mathcal{H}}_s\boldsymbol{C}_{2e,N}\right)\boldsymbol{R}_{\theta\theta}\left(\boldsymbol{C}_{2e,N}^{-1}\boldsymbol{\mathcal{H}}_s\boldsymbol{C}_{2e,N}\right)^T\\
&+\left(-\boldsymbol{C}_{2e,N}^{-1}\boldsymbol{\mathcal{F}}_a\boldsymbol{S}_{2e,N}\right)\boldsymbol{R}_{ww}\boldsymbol{S}_{2e,N}^{T}\boldsymbol{S}_{2e,N}^{-T}+\left(\boldsymbol{C}_{2e,N}^{-1}\boldsymbol{\mathcal{F}}_s\boldsymbol{C}_{2e,N}\right)\boldsymbol{R}_{ww}\boldsymbol{C}_{2e,N}^{T}\boldsymbol{C}_{2e,N}^{-T}\\
&=\boldsymbol{C}_{2e,N}^{-1}\boldsymbol{C}_{2e,N}\boldsymbol{R}_{\theta\theta}\left(\boldsymbol{S}_{2e,N}^{-1}\boldsymbol{\mathcal{H}}_a\boldsymbol{C}_{2e,N}\right)^T+\boldsymbol{C}_{2e,N}^{-1}\boldsymbol{C}_{2e,N}\boldsymbol{R}_{\theta\theta}\left(\boldsymbol{C}_{2e,N}^{-1}\boldsymbol{\mathcal{H}}_s\boldsymbol{C}_{2e,N}\right)^T. \qquad (116)
\end{aligned}
$$

Equation (116) simplifies to

$$
\begin{aligned}
&\boldsymbol{C}_{2e,N}^{-1}\left[-\boldsymbol{\mathcal{F}}_a\left(\boldsymbol{\mathcal{H}}_a\boldsymbol{R}_{\Theta_s\Theta_s}\boldsymbol{\mathcal{H}}_a+\boldsymbol{R}_{w_aw_a}\right)+\boldsymbol{\mathcal{F}}_s\boldsymbol{\mathcal{H}}_s\boldsymbol{R}_{\Theta_s\Theta_s}\boldsymbol{\mathcal{H}}_a\right]\boldsymbol{S}_{2e,N}^{-T}\\
&+\boldsymbol{C}_{2e,N}^{-1}\left[-\boldsymbol{\mathcal{F}}_a\boldsymbol{\mathcal{H}}_a\boldsymbol{R}_{\Theta_s\Theta_s}\boldsymbol{\mathcal{H}}_s+\boldsymbol{\mathcal{F}}_s\left(\boldsymbol{\mathcal{H}}_s\boldsymbol{R}_{\Theta_s\Theta_s}\boldsymbol{\mathcal{H}}_s+\boldsymbol{R}_{w_sw_s}\right)\right]\boldsymbol{C}_{2e,N}^{-T}\\
&+\boldsymbol{S}_{2e,N}^{-1}\left[\boldsymbol{\mathcal{F}}_a\boldsymbol{\mathcal{H}}_s\boldsymbol{R}_{\Theta_s\Theta_s}\boldsymbol{\mathcal{H}}_a+\boldsymbol{\mathcal{F}}_s\boldsymbol{\mathcal{H}}_a\boldsymbol{R}_{\Theta_s\Theta_s}\boldsymbol{\mathcal{H}}_a\right]\boldsymbol{S}_{2e,N}^{-T}\\
&+\boldsymbol{S}_{2e,N}^{-1}\left[\boldsymbol{\mathcal{F}}_a\boldsymbol{\mathcal{H}}_s\boldsymbol{R}_{\Theta_s\Theta_s}\boldsymbol{\mathcal{H}}_s+\boldsymbol{\mathcal{F}}_s\boldsymbol{\mathcal{H}}_a\boldsymbol{R}_{\Theta_s\Theta_s}\boldsymbol{\mathcal{H}}_s\right]\boldsymbol{C}_{2e,N}^{-T}\\
&=\boldsymbol{C}_{2e,N}^{-1}\boldsymbol{R}_{\Theta_s\Theta_s}\boldsymbol{\mathcal{H}}_a\boldsymbol{S}_{2e,N}^{-T}+\boldsymbol{C}_{2e,N}^{-1}\boldsymbol{R}_{\Theta_s\Theta_s}\boldsymbol{\mathcal{H}}_s\boldsymbol{C}_{2e,N}^{-T}, \qquad (117)
\end{aligned}
$$

where $\boldsymbol{R}_{\Theta_s\Theta_s}=\boldsymbol{C}_{2e,N}\boldsymbol{R}_{\theta\theta}\boldsymbol{C}_{2e,N}^{T}$, $\boldsymbol{R}_{w_aw_a}=\boldsymbol{S}_{2e,N}\boldsymbol{R}_{ww}\boldsymbol{S}_{2e,N}^{T}$, and $\boldsymbol{R}_{w_sw_s}=\boldsymbol{C}_{2e,N}\boldsymbol{R}_{ww}\boldsymbol{C}_{2e,N}^{T}$. The simplification of Eq. (116) to produce Eq. (117) uses the facts that $\boldsymbol{\mathcal{H}}_a=\boldsymbol{\mathcal{H}}_a^T$ and $\boldsymbol{\mathcal{H}}_s=\boldsymbol{\mathcal{H}}_s^T$. For Eq. (117) to hold, it must follow that

$$
-\boldsymbol{\mathcal{F}}_a\left(\boldsymbol{\mathcal{H}}_a\boldsymbol{R}_{\Theta_s\Theta_s}\boldsymbol{\mathcal{H}}_a+\boldsymbol{R}_{w_aw_a}\right)+\boldsymbol{\mathcal{F}}_s\boldsymbol{\mathcal{H}}_s\boldsymbol{R}_{\Theta_s\Theta_s}\boldsymbol{\mathcal{H}}_a=\boldsymbol{R}_{\Theta_s\Theta_s}\boldsymbol{\mathcal{H}}_a, \qquad (118)
$$

$$
-\boldsymbol{\mathcal{F}}_a\boldsymbol{\mathcal{H}}_a\boldsymbol{R}_{\Theta_s\Theta_s}\boldsymbol{\mathcal{H}}_s+\boldsymbol{\mathcal{F}}_s\left(\boldsymbol{\mathcal{H}}_s\boldsymbol{R}_{\Theta_s\Theta_s}\boldsymbol{\mathcal{H}}_s+\boldsymbol{R}_{w_sw_s}\right)=\boldsymbol{R}_{\Theta_s\Theta_s}\boldsymbol{\mathcal{H}}_s, \qquad (119)
$$

$$
\boldsymbol{\mathcal{F}}_a\boldsymbol{\mathcal{H}}_s\boldsymbol{R}_{\Theta_s\Theta_s}\boldsymbol{\mathcal{H}}_a+\boldsymbol{\mathcal{F}}_s\boldsymbol{\mathcal{H}}_a\boldsymbol{R}_{\Theta_s\Theta_s}\boldsymbol{\mathcal{H}}_a=\boldsymbol{0}, \qquad (120)
$$

and

$$
\boldsymbol{\mathcal{F}}_a\boldsymbol{\mathcal{H}}_s\boldsymbol{R}_{\Theta_s\Theta_s}\boldsymbol{\mathcal{H}}_s+\boldsymbol{\mathcal{F}}_s\boldsymbol{\mathcal{H}}_a\boldsymbol{R}_{\Theta_s\Theta_s}\boldsymbol{\mathcal{H}}_s=\boldsymbol{0}. \qquad (121)
$$

Every matrix in Eqs. (118) - (121) is either exactly or approximately diagonal. Recognize that the matrices $\boldsymbol{\mathcal{F}}_a$, $\boldsymbol{\mathcal{F}}_s$, $\boldsymbol{\mathcal{H}}_a$, and $\boldsymbol{\mathcal{H}}_s$ are always diagonal because they are the diagonalized forms of symmetric convolution matrices in the trigonometric transform domain. The matrices $\boldsymbol{R}_{w_aw_a}$ and $\boldsymbol{R}_{w_sw_s}$ will be diagonal because the samples of $\boldsymbol{w}$ are assumed to be uncorrelated with

each other and to have uniform variance, $\sigma^2$, so that $\boldsymbol{R}_{ww} = \sigma^2 \boldsymbol{I}$. In general, the transform-domain correlation matrix $\boldsymbol{R}_{\Theta_s\Theta_s}$ will be well-approximated by its diagonal elements if the samples of the object, $\theta$, are highly correlated with each other [27]. Specifically if $\boldsymbol{R}_{\theta\theta}$ is the $N \times N$ correlation matrix for a Markov-I process, [38], [47], [53], then the correlation matrix $\boldsymbol{R}_{\Theta_s\Theta_s}$ in the DCT domain will approximate a diagonal matrix. For a Markov-I process, the $m-n$th element of $\boldsymbol{R}_{\theta\theta}$ is $\left[\boldsymbol{R}_{\theta\theta}\right]_{mn} = \rho^{|m-n|}$ where $0 < \rho < 1$. As an example of how well the transform domain correlation matrix approximates a diagonal matrix, let $\boldsymbol{R}_{\theta\theta}$ be the $256 \times 256$ correlation matrix of a Markov-I process with $\rho = 0.9$. Then the matrix $\boldsymbol{R}_{\Theta_s\Theta_s} = \boldsymbol{C}_{2e,N}\boldsymbol{R}_{\theta\theta}\boldsymbol{C}_{2e,N}^T$ in the DCT domain has 98.75% of its total energy, *i.e.*, the sum of the squares of its elements, along the diagonal. As a comparison the Fourier domain matrix $\boldsymbol{R}_{\Theta_F\Theta_F} = \boldsymbol{W}_N^{-1}\boldsymbol{R}_{\theta\theta}\boldsymbol{W}_N^{-H}$ has 98.20% of its total energy contained in its diagonal elements. The transform domain correlation matrix, $\boldsymbol{R}_{\Theta_s\Theta_s}$, will approach an exactly diagonal matrix either as $N$ increases without bound or as $\rho$ increases to 1. This approximate diagonalization occurs because the DCT provides an excellent approximation to the eigenvectors of the correlation matrix of a Markov-I process [27], [39]. Thus under the above assumptions for the noise and object processes, $\boldsymbol{R}_{\mathcal{W}_a\mathcal{W}_a}$ and $\boldsymbol{R}_{\mathcal{W}_s\mathcal{W}_s}$ are exactly represented by their diagonal elements $\overline{\mathcal{W}_a^2(k)}$ and $\overline{\mathcal{W}_s^2(k)}$ for $k = 0, 1, \ldots, N$, and $\boldsymbol{R}_{\Theta_s\Theta_s}$ is well-approximated by its diagonal elements $\overline{\Theta_s^2(k)}$ for $k = 0, 1, \ldots, N-1$.

The above approximations allow for a scalar solution based solely on the diagonal elements of the matrices in Eqs. (118) - (121). Solving for the diagonal elements of Eqs. (118) - (121) produces the $4 \times 2$ overdetermined matrix equation

$$\begin{bmatrix} \mathcal{H}_a^2(k)+\dfrac{\overline{\mathcal{W}_a^2(k)}}{\Theta_s^2(k)} & -\mathcal{H}_a(k)\mathcal{H}_s(k) \\ -\mathcal{H}_a(k)\mathcal{H}_s(k) & \mathcal{H}_s^2(k)+\dfrac{\overline{\mathcal{W}_s^2(k)}}{\Theta_s^2(k)} \\ \mathcal{H}_a(k)\mathcal{H}_s(k) & \mathcal{H}_a^2(k) \\ \mathcal{H}_s^2(k) & \mathcal{H}_a(k)\mathcal{H}_s(k) \end{bmatrix} \begin{bmatrix} \mathcal{F}_a(k) \\ \mathcal{F}_s(k) \end{bmatrix} = \begin{bmatrix} -\mathcal{H}_a(k) \\ \mathcal{H}_s(k) \\ 0 \\ 0 \end{bmatrix}. \tag{122}$$

Equation (122) reduces to

$$\begin{bmatrix} \mathcal{H}_a^2+\mathcal{H}_s^2+\dfrac{\overline{\mathcal{W}_a^2}}{\Theta_s^2} & 0 \\ 0 & \mathcal{H}_a^2+\mathcal{H}_s^2+\dfrac{\overline{\mathcal{W}_s^2}}{\Theta_s^2} \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \mathcal{F}_a \\ \mathcal{F}_s \end{bmatrix} = \begin{bmatrix} -\mathcal{H}_a \\ \mathcal{H}_s \\ \dfrac{\mathcal{H}_a\mathcal{H}_s}{\mathcal{H}_a^2+\mathcal{H}_s^2+\dfrac{\overline{\mathcal{W}_a^2}}{\Theta_s^2}}+\dfrac{-\mathcal{H}_a\mathcal{H}_s}{\mathcal{H}_a^2+\mathcal{H}_s^2+\dfrac{\overline{\mathcal{W}_s^2}}{\Theta_s^2}} \\ 0 \end{bmatrix}, \tag{123}$$

where each term in Eq. (123) depends on the index, $k$, but temporarily has the indexing suppressed to preserve space in the equation. The system of equations will be consistent, and Eq. (123) will equal

$$\begin{bmatrix} \mathcal{H}_a^2(k)+\mathcal{H}_s^2(k)+\dfrac{\overline{\mathcal{W}_a^2(k)}}{\Theta_s^2(k)} & 0 \\ 0 & \mathcal{H}_a^2(k)+\mathcal{H}_s^2(k)+\dfrac{\overline{\mathcal{W}_s^2(k)}}{\Theta_s^2(k)} \end{bmatrix} \begin{bmatrix} \mathcal{F}_a(k) \\ \mathcal{F}_s(k) \end{bmatrix} = \begin{bmatrix} -\mathcal{H}_a(k) \\ \mathcal{H}_s(k) \end{bmatrix}, \tag{124}$$

if

$$\frac{\mathcal{H}_a(k)\mathcal{H}_s(k)}{\mathcal{H}_a^2(k)+\mathcal{H}_s^2(k)+\dfrac{\overline{\mathcal{W}_a^2(k)}}{\overline{\Theta_s^2(k)}}} = \frac{\mathcal{H}_a(k)\mathcal{H}_s(k)}{\mathcal{H}_a^2(k)+\mathcal{H}_s^2(k)+\dfrac{\overline{\mathcal{W}_s^2(k)}}{\overline{\Theta_s^2(k)}}}. \tag{125}$$

It is straightforward from the definitions of the DTTs and the definitions of $\boldsymbol{R}_{w_a w_a}$ and $\boldsymbol{R}_{w_s w_s}$ to show that

$$\overline{\mathcal{W}_a^2(k)} = \begin{cases} 0, & k = 0 \\ 2N\sigma^2, & k = 1, 2, \ldots, N-1 \\ 4N\sigma^2, & k = N \end{cases} \tag{126}$$

and

$$\overline{\mathcal{W}_s^2(k)} = \begin{cases} 4N\sigma^2, & k = 0 \\ 2N\sigma^2, & k = 1, 2, \ldots, N-1 \\ 0, & k = N \end{cases} \tag{127}$$

whenever $\boldsymbol{R}_{ww} = \sigma^2 \boldsymbol{I}$. From Eqs. (126) and (127) it follows that Eq. (125) holds for $k = 1, 2, \ldots, N-1$. It is also straightforward to verify Eq. (125) at $k = 0$ by using the fact that $\mathcal{H}_a(0) = 0$ based on its required symmetry in the transform domain [34]. Verifying that Eq. (125) holds at $k = N$ requires the facts that $\overline{\Theta_s^2(N)} = 0$ and $\mathcal{H}_a(N) = 0$ based on their required symmetry in the transform domain [34]. Then L'Hôpital's rule must be applied to take the limit as $\overline{\Theta_s^2(N)} \to 0$ of the derivative of the numerator and denominator of each side of Eq. (125) with respect to $\overline{\Theta_s^2(N)}$.

Thus Eq. (125) is valid for all values of $k$, and Eq. (124) will hold, from which matrix multiplication followed by scalar division produces

$$\mathcal{F}_a(k) = \frac{-\mathcal{H}_a(k)}{\mathcal{H}_a^2(k)+\mathcal{H}_s^2(k)+\dfrac{\overline{\mathcal{W}_a^2(k)}}{\overline{\Theta_s^2(k)}}} \tag{128}$$

and
$$\mathcal{F}_s(k) = \frac{\mathcal{H}_s(k)}{\mathcal{H}_a^2(k) + \mathcal{H}_s^2(k) + \dfrac{\overline{\mathcal{W}_s^2(k)}}{\Theta_s^2(k)}}. \tag{129}$$

Equations (128) and (129) are the one-dimensional scalar Wiener filter expressed in the trigonometric transform domain for symmetrically convolved sequences. Note first that for the noise-free case where $\boldsymbol{w} = \mathbf{0}$, which implies that $\overline{\mathcal{W}_a^2(k)} = 0$ and $\overline{\mathcal{W}_s^2(k)} = 0$, Eqs. (128) and (129) reduce to Eqs. (97) and (98), and the scalar Wiener filter becomes an inverse filter. Note also that if the sequence $h(n)$ possesses strictly whole-sample symmetry, *i.e.*, $h_a(n) = \mathcal{H}_a(k) = 0$, then the scalar Wiener filter reduces to

$$\mathcal{F}_s(k) = \frac{\mathcal{H}_s(k)}{\mathcal{H}_s^2(k) + \dfrac{\overline{\mathcal{W}_s^2(k)}}{\Theta_s^2(k)}}. \tag{130}$$

Equation (130) closely resembles Eq. (23) for the one-dimensional Fourier case. Similar results for two dimensions are shown in the following subsection.

**4.3.2** ***The Two-Dimensional Scalar Wiener Filter for Trigonometric Transforms.*** The derivation of the two-dimensional scalar Wiener filter in the trigonometric transform domain is somewhat more involved than it was for one dimension. The two-dimensional version of the general asymmetric cases for the filter matrices requires a four-way decomposition for each block symmetric convolution matrix, so that $\boldsymbol{F} = \boldsymbol{F}_{aa} + \boldsymbol{F}_{as} + \boldsymbol{F}_{sa} + \boldsymbol{F}_{ss}$ and $\boldsymbol{H} = \boldsymbol{H}_{aa} + \boldsymbol{H}_{as} + \boldsymbol{H}_{sa} + \boldsymbol{H}_{ss}$. Making these substitutions into Eq. (114) yields

$$\begin{aligned}\left(\boldsymbol{F}_{aa} + \boldsymbol{F}_{as} + \boldsymbol{F}_{sa} + \boldsymbol{F}_{ss}\right)\Big[\left(\boldsymbol{H}_{aa} + \boldsymbol{H}_{as} + \boldsymbol{H}_{sa} + \boldsymbol{H}_{ss}\right)\boldsymbol{R}_{\theta\theta}\left(\boldsymbol{H}_{aa} + \boldsymbol{H}_{as} + \boldsymbol{H}_{sa} + \boldsymbol{H}_{ss}\right)^T + \boldsymbol{R}_{ww}\Big] \\ = \boldsymbol{R}_{\theta\theta}\left(\boldsymbol{H}_{aa} + \boldsymbol{H}_{as} + \boldsymbol{H}_{aa} + \boldsymbol{H}_{as}\right)^T.\end{aligned} \tag{131}$$

The expansion of Eq. (131) produces 68 terms on the left-hand side. The Kronecker product transformations of the convolution matrices in Eqs. (76) - (91) must then transform each of these

terms into the trigonometric transform domain. The two-dimensional versions of Eqs. (116) - (121) will subsequently become quite cumbersome. The details of this lengthy derivation appear in the appendix. The end result in the appendix is a $16 \times 4$ overdetermined system of equations similar to Eq. (122), which reduces to the $7 \times 4$ matrix equation

$$\begin{bmatrix} \mathcal{H}_{aa}^2 + \dfrac{\overline{\mathcal{W}_{aa}^2}}{\Theta_{ss}^2} & -\mathcal{H}_{aa}\mathcal{H}_{as} & -\mathcal{H}_{aa}\mathcal{H}_{sa} & \mathcal{H}_{aa}\mathcal{H}_{ss} \\ -\mathcal{H}_{aa}\mathcal{H}_{as} & \mathcal{H}_{as}^2 + \dfrac{\overline{\mathcal{W}_{as}^2}}{\Theta_{ss}^2} & \mathcal{H}_{as}\mathcal{H}_{sa} & -\mathcal{H}_{as}\mathcal{H}_{ss} \\ -\mathcal{H}_{aa}\mathcal{H}_{sa} & \mathcal{H}_{as}\mathcal{H}_{sa} & \mathcal{H}_{sa}^2 + \dfrac{\overline{\mathcal{W}_{sa}^2}}{\Theta_{ss}^2} & -\mathcal{H}_{sa}\mathcal{H}_{ss} \\ \mathcal{H}_{aa}\mathcal{H}_{ss} & -\mathcal{H}_{as}\mathcal{H}_{ss} & -\mathcal{H}_{sa}\mathcal{H}_{ss} & \mathcal{H}_{ss}^2 + \dfrac{\overline{\mathcal{W}_{ss}^2}}{\Theta_{ss}^2} \\ \mathcal{H}_{as} & \mathcal{H}_{aa} & -\mathcal{H}_{ss} & -\mathcal{H}_{sa} \\ \mathcal{H}_{sa} & -\mathcal{H}_{ss} & \mathcal{H}_{aa} & -\mathcal{H}_{as} \\ \mathcal{H}_{ss} & \mathcal{H}_{sa} & \mathcal{H}_{as} & \mathcal{H}_{aa} \end{bmatrix} \begin{bmatrix} \mathcal{F}_{aa} \\ \mathcal{F}_{as} \\ \mathcal{F}_{sa} \\ \mathcal{F}_{ss} \end{bmatrix} = \begin{bmatrix} \mathcal{H}_{aa} \\ -\mathcal{H}_{as} \\ -\mathcal{H}_{sa} \\ \mathcal{H}_{ss} \\ 0 \\ 0 \\ 0 \end{bmatrix}. \tag{132}$$

All the quantities in the matrix equation are indexed over $k_1$ and $k_2$, but Eq. (132) temporarily suppresses the indexing to conserve space.

Unfortunately the set of equations in Eq. (132) is not consistent as it was in the one-dimensional case. Thus no general solution exists in the trigonometric transform domain for the general case of a PSF, $h(n_1,n_2)$, which possesses all four types of underlying symmetry. Solutions exist for $f(n_1,n_2)$ in the sequence domain from solving Eq. (131) directly, but this involves a large matrix inversion for even small image and PSF sizes. Transform domain representations

are computationally easier to implement, so the following discussion focuses on the conditions under which a solution to Eq. (132) exists.

The solution $\mathcal{F}_{aa}(k_1,k_2) = \mathcal{F}_{as}(k_1,k_2) = \mathcal{F}_{sa}(k_1,k_2) = \mathcal{F}_{ss}(k_1,k_2) = 0$ exists trivially for the case when $\mathcal{H}_{aa}(k_1,k_2) = \mathcal{H}_{as}(k_1,k_2) = \mathcal{H}_{sa}(k_1,k_2) = \mathcal{H}_{ss}(k_1,k_2) = 0$, but is of little use. A solution also exists for the four cases where any three of the four components are zero. If $\mathcal{H}_{as}(k_1,k_2) = \mathcal{H}_{sa}(k_1,k_2) = \mathcal{H}_{ss}(k_1,k_2) = 0$ and $\mathcal{H}_{aa}(k_1,k_2) \neq 0$, then the PSF $h(n_1,n_2)$ would possess strictly whole-sample antisymmetry about both the $n_1$ and $n_2$ axes, and the scalar Wiener filter for this case is

$$\mathcal{F}_{aa}(k_1,k_2) = \frac{\mathcal{H}_{aa}(k_1,k_2)}{\mathcal{H}_{aa}^2(k_1,k_2) + \dfrac{\mathcal{W}_{aa}^2(k_1,k_2)}{\Theta_{ss}^2(k_1,k_2)}}. \tag{133}$$

For the case where $\mathcal{H}_{aa}(k_1,k_2) = \mathcal{H}_{sa}(k_1,k_2) = \mathcal{H}_{ss}(k_1,k_2) = 0$ and $\mathcal{H}_{as}(k_1,k_2) \neq 0$, the scalar Wiener filter is

$$\mathcal{F}_{as}(k_1,k_2) = \frac{-\mathcal{H}_{as}(k_1,k_2)}{\mathcal{H}_{as}^2(k_1,k_2) + \dfrac{\mathcal{W}_{as}^2(k_1,k_2)}{\Theta_{ss}^2(k_1,k_2)}}. \tag{134}$$

For the case where $\mathcal{H}_{aa}(k_1,k_2) = \mathcal{H}_{as}(k_1,k_2) = \mathcal{H}_{ss}(k_1,k_2) = 0$ and $\mathcal{H}_{sa}(k_1,k_2) \neq 0$, the scalar Wiener filter is

$$\mathcal{F}_{sa}(k_1,k_2) = \frac{-\mathcal{H}_{sa}(k_1,k_2)}{\mathcal{H}_{sa}^2(k_1,k_2) + \dfrac{\mathcal{W}_{sa}^2(k_1,k_2)}{\Theta_{ss}^2(k_1,k_2)}}. \tag{135}$$

For the case where $\mathcal{H}_{aa}(k_1,k_2) = \mathcal{H}_{as}(k_1,k_2) = \mathcal{H}_{sa}(k_1,k_2) = 0$ and $\mathcal{H}_{ss}(k_1,k_2) \neq 0$, the scalar Wiener filter is

$$\mathcal{F}_{ss}(k_1,k_2)=\frac{\mathcal{H}_{ss}(k_1,k_2)}{\mathcal{H}_{ss}^2(k_1,k_2)+\dfrac{\overline{\mathcal{W}_{ss}^2(k_1,k_2)}}{\overline{\Theta_{ss}^2(k_1,k_2)}}}. \tag{136}$$

Equation (136) is the scalar Wiener filter for the case of a PSF which possesses strictly whole sample symmetry about both the $n_1$ and $n_2$ axes.

Note that the subscript *'ss'* remains in the term $\overline{\Theta_{ss}^2(k_1,k_2)}$ for all four cases in Eqs. (133) - (136). Throughout the derivation of the two-dimensional scalar Wiener filter presented in the appendix, substitutions of equivalent convolution matrices in the transform domain from Eqs. (76) - (91) has resulted in a transform domain object correlation matrix of the form $\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}=\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\boldsymbol{R}_{\theta\theta}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^T$. Choosing transform relations so that the transform domain object correlation matrix always lies in the transform domain of the type-II DCT is what allows it to be well-approximated by its diagonal elements and yield a scalar filter.

Equation (132) will also produce a consistent set of equations in four of the six cases where two of the four types of underlying symmetry are simultaneously nonzero. The equations will be inconsistent for the two cases where:

(*i*) $\mathcal{H}_{aa}(k_1,k_2)=\mathcal{H}_{ss}(k_1,k_2)=0,\ \mathcal{H}_{as}(k_1,k_2)\neq 0,$ and $\mathcal{H}_{sa}(k_1,k_2)\neq 0;$

and (*ii*) $\mathcal{H}_{as}(k_1,k_2)=\mathcal{H}_{sa}(k_1,k_2)=0,\ \mathcal{H}_{aa}(k_1,k_2)\neq 0,$ and $\mathcal{H}_{ss}(k_1,k_2)\neq 0.$

They will be consistent for the four cases where:

(*i*) $\mathcal{H}_{aa}(k_1,k_2)=\mathcal{H}_{as}(k_1,k_2)=0,\ \mathcal{H}_{sa}(k_1,k_2)\neq 0,$ and $\mathcal{H}_{ss}(k_1,k_2)\neq 0;$

(*ii*) $\mathcal{H}_{aa}(k_1,k_2)=\mathcal{H}_{sa}(k_1,k_2)=0,\ \mathcal{H}_{as}(k_1,k_2)\neq 0,$ and $\mathcal{H}_{ss}(k_1,k_2)\neq 0;$

(*iii*) $\mathcal{H}_{as}(k_1,k_2)=\mathcal{H}_{ss}(k_1,k_2)=0,\ \mathcal{H}_{aa}(k_1,k_2)\neq 0,$ and $\mathcal{H}_{sa}(k_1,k_2)\neq 0;$

and (*iv*) $\mathcal{H}_{sa}(k_1,k_2)=\mathcal{H}_{ss}(k_1,k_2)=0,\ \mathcal{H}_{aa}(k_1,k_2)\neq 0,$ and $\mathcal{H}_{as}(k_1,k_2)\neq 0.$

For the first case where $\mathcal{H}_{aa}(k_1,k_2)=\mathcal{H}_{as}(k_1,k_2)=0$, $\mathcal{H}_{sa}(k_1,k_2)\neq 0$, and $\mathcal{H}_{ss}(k_1,k_2)\neq 0$, the resulting scalar Wiener filter will be

$$\mathcal{F}_{aa}(k_1,k_2)=0, \tag{137}$$

$$\mathcal{F}_{as}(k_1,k_2)=0, \tag{138}$$

$$\mathcal{F}_{sa}(k_1,k_2)=\frac{-\mathcal{H}_{sa}(k_1,k_2)}{\mathcal{H}_{sa}^2(k_1,k_2)+\mathcal{H}_{ss}^2(k_1,k_2)+\dfrac{\overline{\mathcal{W}_{sa}^2(k_1,k_2)}}{\overline{\Theta_{ss}^2(k_1,k_2)}}}, \tag{139}$$

and

$$\mathcal{F}_{ss}(k_1,k_2)=\frac{\mathcal{H}_{ss}(k_1,k_2)}{\mathcal{H}_{sa}^2(k_1,k_2)+\mathcal{H}_{ss}^2(k_1,k_2)+\dfrac{\overline{\mathcal{W}_{ss}^2(k_1,k_2)}}{\overline{\Theta_{ss}^2(k_1,k_2)}}}. \tag{140}$$

For the second case where $\mathcal{H}_{aa}(k_1,k_2)=\mathcal{H}_{sa}(k_1,k_2)=0$, $\mathcal{H}_{as}(k_1,k_2)\neq 0$, and $\mathcal{H}_{ss}(k_1,k_2)\neq 0$, the resulting scalar Wiener filter will be

$$\mathcal{F}_{aa}(k_1,k_2)=0, \tag{141}$$

$$\mathcal{F}_{as}(k_1,k_2)=\frac{-\mathcal{H}_{as}(k_1,k_2)}{\mathcal{H}_{as}^2(k_1,k_2)+\mathcal{H}_{ss}^2(k_1,k_2)+\dfrac{\overline{\mathcal{W}_{as}^2(k_1,k_2)}}{\overline{\Theta_{ss}^2(k_1,k_2)}}}, \tag{142}$$

$$\mathcal{F}_{sa}(k_1,k_2)=0, \tag{143}$$

and

$$\mathcal{F}_{ss}(k_1,k_2)=\frac{\mathcal{H}_{ss}(k_1,k_2)}{\mathcal{H}_{as}^2(k_1,k_2)+\mathcal{H}_{ss}^2(k_1,k_2)+\dfrac{\overline{\mathcal{W}_{ss}^2(k_1,k_2)}}{\overline{\Theta_{ss}^2(k_1,k_2)}}}. \tag{144}$$

For the third case where $\mathcal{H}_{as}(k_1,k_2)=\mathcal{H}_{ss}(k_1,k_2)=0$, $\mathcal{H}_{aa}(k_1,k_2)\neq 0$, and $\mathcal{H}_{sa}(k_1,k_2)\neq 0$, the resulting scalar Wiener filter will be

$$\mathcal{F}_{aa}(k_1,k_2)=\frac{\mathcal{H}_{aa}(k_1,k_2)}{\mathcal{H}_{aa}^2(k_1,k_2)+\mathcal{H}_{sa}^2(k_1,k_2)+\dfrac{\overline{\mathcal{W}_{aa}^2(k_1,k_2)}}{\overline{\Theta_{ss}^2(k_1,k_2)}}}, \tag{145}$$

$$\mathcal{F}_{as}(k_1,k_2)=0, \tag{146}$$

$$\mathcal{F}_{sa}(k_1,k_2)=\frac{-\mathcal{H}_{sa}(k_1,k_2)}{\mathcal{H}_{aa}^2(k_1,k_2)+\mathcal{H}_{sa}^2(k_1,k_2)+\dfrac{\mathcal{W}_{sa}^2(k_1,k_2)}{\mathcal{O}_{ss}^2(k_1,k_2)}}, \tag{147}$$

and

$$\mathcal{F}_{ss}(k_1,k_2)=0. \tag{148}$$

For the fourth case where $\mathcal{H}_{sa}(k_1,k_2)=\mathcal{H}_{ss}(k_1,k_2)=0$, $\mathcal{H}_{aa}(k_1,k_2)\neq 0$, and $\mathcal{H}_{as}(k_1,k_2)\neq 0$, the resulting scalar Wiener filter will be

$$\mathcal{F}_{aa}(k_1,k_2)=\frac{\mathcal{H}_{aa}(k_1,k_2)}{\mathcal{H}_{aa}^2(k_1,k_2)+\mathcal{H}_{as}^2(k_1,k_2)+\dfrac{\mathcal{W}_{aa}^2(k_1,k_2)}{\mathcal{O}_{ss}^2(k_1,k_2)}}, \tag{149}$$

$$\mathcal{F}_{as}(k_1,k_2)=\frac{-\mathcal{H}_{as}(k_1,k_2)}{\mathcal{H}_{aa}^2(k_1,k_2)+\mathcal{H}_{as}^2(k_1,k_2)+\dfrac{\mathcal{W}_{as}^2(k_1,k_2)}{\mathcal{O}_{ss}^2(k_1,k_2)}}, \tag{150}$$

$$\mathcal{F}_{sa}(k_1,k_2)=0, \tag{151}$$

and

$$\mathcal{F}_{ss}(k_1,k_2)=0. \tag{152}$$

Note that for each of the eight nontrivial cases where the two-dimensional scalar Wiener filter exists, it reduces to the inverse filter of Eqs. (107) - (111) whenever the noise terms go to zero. The set of equations in Eq. (132) is inconsistent for each of the four cases having three nonzero terms and for the general case of four nonzero terms.

Two options are available to find a solution to the general problem of a PSF having all four types of underlying symmetry. The first option finds the least-squares solution to the over-determined system in Eq. (132). The second option makes different substitutions from equivalent forms for the transforms to yield a $4\times 4$ system of equations.

Using shorthand notation to express the $7 \times 4$ matrix equation in Eq. (132) as $\mathcal{H}\mathcal{f} = \mathcal{g}$, the least-squares solution is $\hat{\mathcal{f}} = \left(\mathcal{H}^T\mathcal{H}\right)^{-1}\mathcal{H}^T\mathcal{g}$. The inverse of $\mathcal{H}^T\mathcal{H}$ exists because $\mathcal{H}$ has rank four. The vector $\hat{\mathcal{f}}$ is the solution which minimizes $\left\|\mathcal{g} - \mathcal{H}\mathcal{f}\right\|_2$ [46]. Unfortunately, both the least squares solution and the solution to the $4 \times 4$ system of equations which results from choosing different equivalent forms for the transforms are too lengthy to be presented.

Each of these two methods of finding a general solution has other disadvantages as well. The disadvantage to finding the least squares solution to Eq. (132) is that it is not exact and will therefore introduce more error in addition to the errors caused by noise and by the scalar approximation which retains just the diagonal elements of the object correlation matrix. The disadvantage to choosing different transforms in the derivation of the two-dimensional filter is that the result depends on the terms $\overline{\Theta^2_{aa}(k_1,k_2)}$, $\overline{\Theta^2_{as}(k_1,k_2)}$, and $\overline{\Theta^2_{sa}(k_1,k_2)}$, which are all not as well-approximated by their diagonal elements as $\overline{\Theta^2_{ss}(k_1,k_2)}$ [26]. The energy in the off-diagonal terms which is not accounted for in the diagonal approximation will introduce additional error as well.

The fact that no exact general solution exists to Eq. (132) is not as restrictive as it might first appear. The form of the solution appearing in Eq. (136) based on a PSF having only whole-sample symmetry is the form of the filter which will likely prove the most useful in practice, because whole-sample symmetry is a necessary condition for a filter to have linear phase [37]. Many image reconstruction methods model asymmetric PSFs with inherent nonlinear phase, such as those used to model the effects of atmospheric turbulence [40], as random processes. In these cases, even though individual realizations of a PSF are asymmetric, the mean PSF will often be

symmetric so that only whole-sample symmetry will be present in the scalar Wiener filter. Equation (136) is a valid form of the scalar Wiener filter in these cases.

The existence of inverse and scalar Wiener filters in the trigonometric transform domain for one and two dimensions has been demonstrated in this chapter. The two dimensional scalar Wiener filter is limited by the type of symmetry present in the PSF in certain cases. It was claimed that the two-dimensional whole-sample symmetric version is the most useful case for image processing. In the following chapter, the performance of the scalar Wiener filter for this particular case is analyzed.

# V. *Performance of the Scalar Wiener Filter for Trigonometric Transforms*

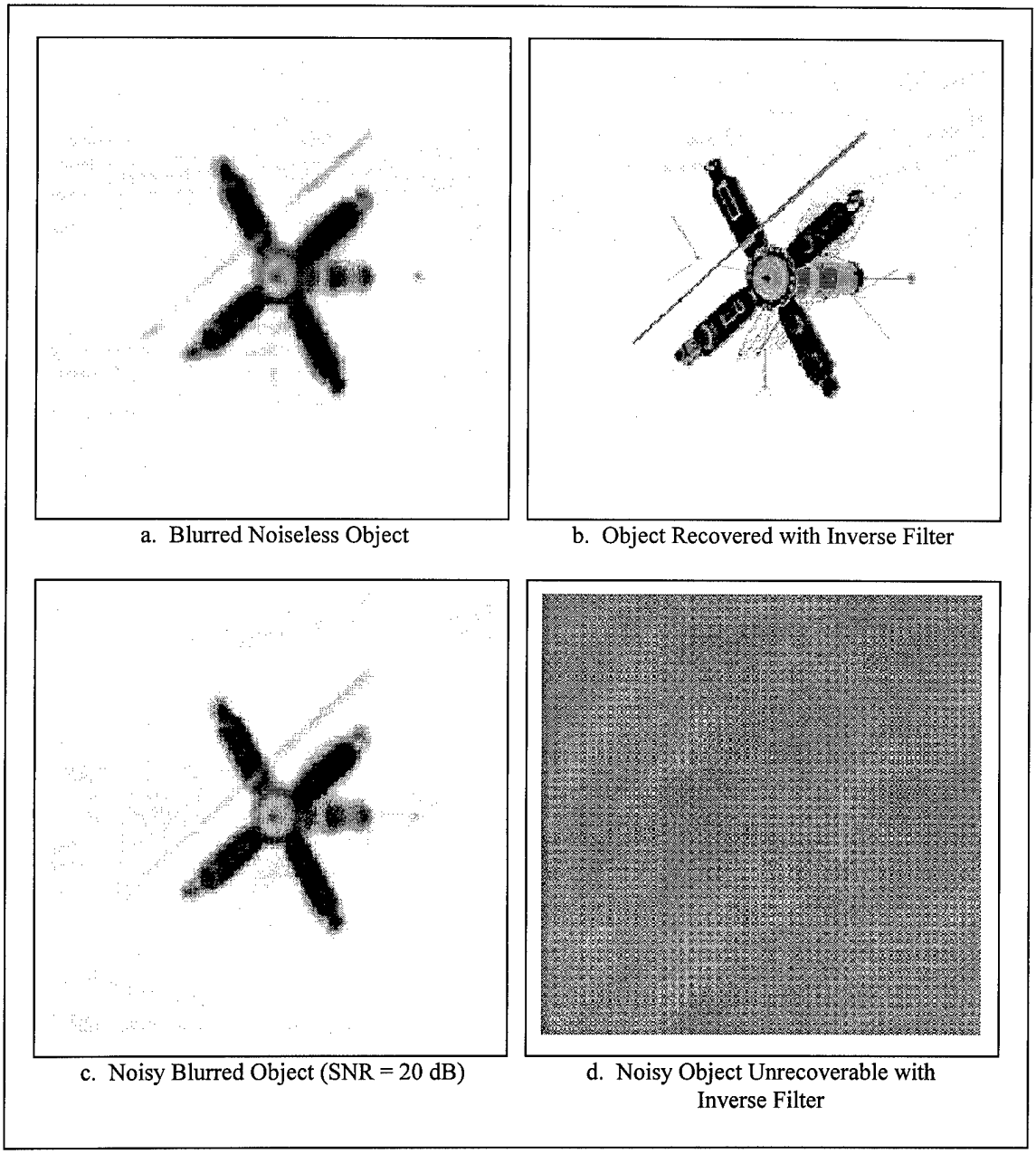
The performance of the two-dimensional scalar Wiener filter is analyzed in this chapter under the assumption that the degradation system point spread function (PSF) is whole-sample symmetric. Systems possessing this characteristic are the most likely to be encountered in practice. Systems with random asymmetric PSFs often have a mean PSF which is whole-sample symmetric. To analyze the scalar Wiener filter's performance, an example is first provided and then the normalized mean-squared error is calculated for several PSFs and objects with varying parameters. The chapter concludes with a brief discussion on the computational advantages of the trigonometric transform-based filter.

## 5.1 *An Example*

In this section, the performance of the new trigonometric transform versions of the two-dimensional inverse and scalar Wiener filters is demonstrated by providing an example. It is shown how a blurred object can be completely recovered with an inverse filter. An inverse filter cannot, however, recover the original object in the presence of noise. A scalar Wiener filter then recovers an estimate of the object from its noisy blurred version by regularizing the high-frequency gain of an inverse filter. The results of this example compare an estimate of the object using a trigonometric scalar Wiener filter to an estimate using a traditional Fourier scalar Wiener filter. This example serves to visually demonstrate the effects of image reconstruction filters in the trigonometric transform domain.

The $256 \times 256$ pixel satellite object shown in Figure 4 for the trigonometric transform domain filtering example in Section 3.4 forms the basis of this example as well. Figure 6 shows

the effects of inverse filtering in the trigonometric transform domain for a blurred noiseless object and a blurred noisy object. In Figure 6(a), the object is blurred using a $16 \times 16$ pixel Gaus-



a. Blurred Noiseless Object

b. Object Recovered with Inverse Filter

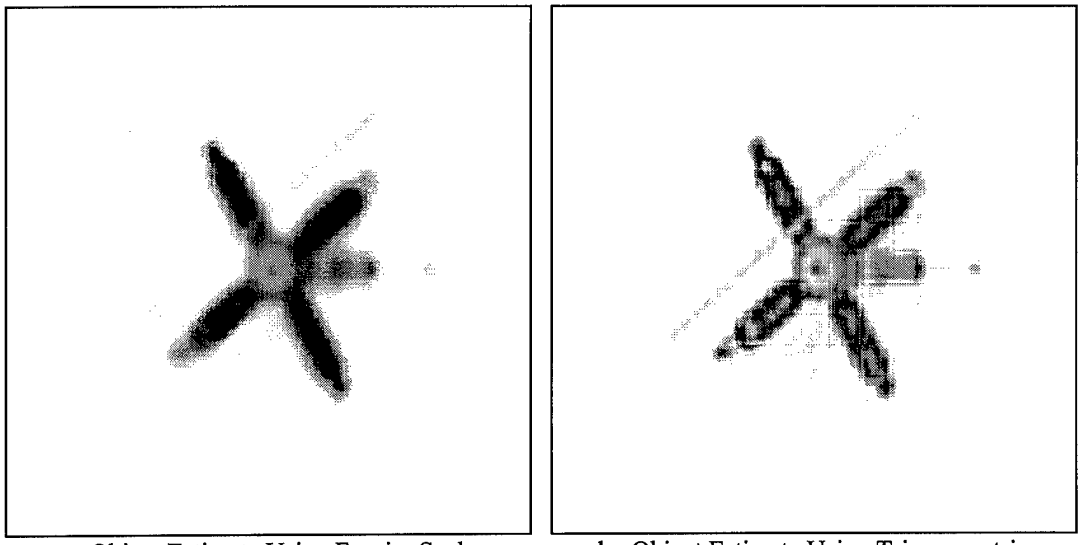
c. Noisy Blurred Object (SNR = 20 dB)

d. Noisy Object Unrecoverable with Inverse Filter

Figure 6. Inverse Trigonometric Filtering Example Using Hanning-Windowed Gaussian Degradation Filter with $1/e$ Width of 4 Pixels

sian-shaped PSF windowed with a Hanning window [37] to prevent an abrupt cutoff at the transition point. The shape of the PSF rolls off to a value of $1/e$ at a distance of 4 pixels from the center. Applying the two-dimensional inverse filter of Eq. (112) in the trigonometric transform domain results in the image shown in Figure 6(b). The original object is completely recovered from its blurred version based on knowledge of the blurring PSF. Figure 6(c) shows the results of adding noise with a signal-to-noise ratio (SNR) of 20 dB to the blurred object. After adding noise, the object is no longer recoverable with an inverse filter as demonstrated by the result in Figure 6(d). The reason the trigonometric inverse filter no longer recovers the object from the noise is that, like the Fourier inverse filter, it amplifies high-frequency components of the noise.

Figure 7 presents the results of applying Fourier and trigonometric scalar Wiener filters to the noisy blurred object of Figure 6(c). Equation (27) implements the scalar Wiener filter in the



a. Object Estimate Using Fourier Scalar Wiener Filter

b. Object Estimate Using Trigonometric Scalar Wiener Filter

Figure 7. Scalar Wiener Filtering Example Using Same Degradation Filter as Inverse Filtering Example

Fourier transform domain, and Eq. (136) implements the scalar Wiener filter in the trigonometric transform domain. Unlike the inverse filters for Figure 6, the scalar Wiener filters for Figure 7 use a block processing technique for their implementation. The need for a block technique arises because the correlation matrices which regularize the inverse filter become very large, even though a scalar Wiener filter is only concerned with their diagonal elements. These diagonal elements in the Fourier domain are the $\overline{\mathcal{W}_F^2(k_1,k_2)}$ and $\overline{\Theta_F^2(k_1,k_2)}$ terms in Eq. (27). The diagonal elements in the trigonometric transform domain are the terms $\overline{\mathcal{W}_{ss}^2(k_1,k_2)}$ and $\overline{\Theta_{ss}^2(k_1,k_2)}$ in Eq. (136). The block processing technique divides the noisy blurred image of Figure 6(c) into $16 \times 16$ blocks, zero pads each block to a size of $32 \times 32$, performs all calculations in $32 \times 32$ tiles of transform space, inverse transforms the results back to the spatial domain, and then sums the overlapping results. Since a Wiener filter depends on an object to originate from a random process with known mean and covariance, this example uses the statistics of the object itself to calculate the matrices $\boldsymbol{R}_{ww}$ and $\boldsymbol{C}_{\theta\theta}$. The object has a nonzero mean which yields a covariance rather than a correlation matrix. The noise variance is set equal to 1/100th of the object variance to achieve a 20 dB SNR.

Comparing Figures 7(a) and (b), it is clear that the trigonometric scalar Wiener filter produces a better quality estimate of the object than the Fourier scalar Wiener filter. One criticism of Wiener filtering is that it tends to oversmooth an image to compensate for noise. This effect is clearly more pronounced in the Fourier case, because the object estimate is still quite blurry although the noise has been averaged out. The trigonometric case achieves a much sharper contrast and higher spatial frequency resolution because of the improved energy compaction of the DCT. Comparing the results of Figures 7(a) and (b) to the original object, a mean-squared error normalized to the total number of pixels is 345.5 for the Fourier case and 211.7 for the trigono-

metric case. Expressed as a ratio to the square of the highest gray scale value of 256, these become -22.7 dB and -24.9 dB, respectively. These values fall within the ranges of some more general expressions for the mean-squared error of the Fourier and trigonometric scalar Wiener filters, as will be seen from the results of the following section.

### 5.2 *Mean-Squared Error Performance*

Closed-form expressions exist for the mean-squared error of Wiener filters because Wiener filtering is a linear technique. Using the method of [28] and [38], the mean-squared error is expressed as $\overline{\|\varepsilon\|_2^2} = \overline{\varepsilon^T \varepsilon} = \mathrm{Tr}\left\{\overline{\varepsilon\varepsilon^T}\right\} = \mathrm{Tr}\left\{\mathrm{E}\left\{\left(\theta - \hat{\theta}\right)\left(\theta - \hat{\theta}\right)^T\right\}\right\}$, which expands to

$$\overline{\|\varepsilon\|_2^2} = \mathrm{Tr}\left\{\boldsymbol{R}_{\theta\theta} - 2\boldsymbol{F}\boldsymbol{H}\boldsymbol{R}_{\theta\theta} + \boldsymbol{F}\left(\boldsymbol{H}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}^T + \boldsymbol{R}_{ww}\right)\boldsymbol{F}^T\right\}. \tag{153}$$

The transform domain expression for Eq. (153) is

$$\overline{\|\boldsymbol{T}\varepsilon\|_2^2} = \mathrm{Tr}\left\{\boldsymbol{R}_{\Theta\Theta} - 2\mathcal{F}\mathcal{H}\boldsymbol{R}_{\Theta\Theta} + \mathcal{F}\left(\mathcal{H}\boldsymbol{R}_{\Theta\Theta}\mathcal{H}^T + \boldsymbol{R}_{\mathcal{W}\mathcal{W}}\right)\mathcal{F}^T\right\}, \tag{154}$$

where the matrix $\boldsymbol{T}$ represents either a DFT or a DCT matrix. All other quantities in Eqs. (153) and (154) have been defined previously. It is easier to calculate the mean-squared error using Eq. (154) in the transform domain because all of the matrices are diagonal except the transform domain correlation matrix, $\boldsymbol{R}_{\Theta\Theta}$. Recall that the diagonal transform domain filter matrix, $\mathcal{F}$, depends on a diagonal approximation of $\boldsymbol{R}_{\Theta\Theta}$. The terms in the diagonal approximation are the $\overline{\Theta_F^2(k_1,k_2)}$ terms from the Fourier domain case of Eq. (27), and the $\overline{\Theta_{ss}^2(k_1,k_2)}$ terms from Eq. (136) for the trigonometric case. To accurately assess the error that this approximation introduces, the error expression must compare the diagonalized version imbedded in the filter matrix, $\mathcal{F}$, to the nondiagonal version in Eq. (154) which it approximates. Each transform has different scale factors which cause different gains in the transform domain. The error calculations

in the transform domain must account for these different scale factors. The error calculations must then apply the appropriate scale factors to calculate the mean-squared error which results back in the spatial domain for both the Fourier and the trigonometric transforms.

Figures 8 - 11 display the results of calculating the error using Eq. (154) in various two-dimensional filtering scenarios. Each filtering scenario depicted uses Hanning-windowed Gaussian-shaped PSF filters to model the degradation similar to the previous example. The scenarios adjust parameters for the length ($N$) of the $N \times N$ PSF, the correlation coefficient ($\rho$) in the Markov-I object correlation matrix, the signal-to-noise ratio (SNR), and the width in pixels to the point at which the PSF has a value of $1/e$. Adjusting these parameters creates different scalar Wiener filters in Eqs. (27) and (136) for the Fourier and trigonometric transform domains. Each graph displays the mean-squared error (MSE) for the Fourier scalar Wiener filter with a solid line, and the MSE for the trigonometric scalar Wiener filter with a dashed line.

In Figure 8, the PSF filter dimension, $N$, increases while the correlation coefficient, $\rho$, is set to 0.9. Additionally the SNR is fixed at 20 dB, and the PSF $1/e$ width increases at a constant

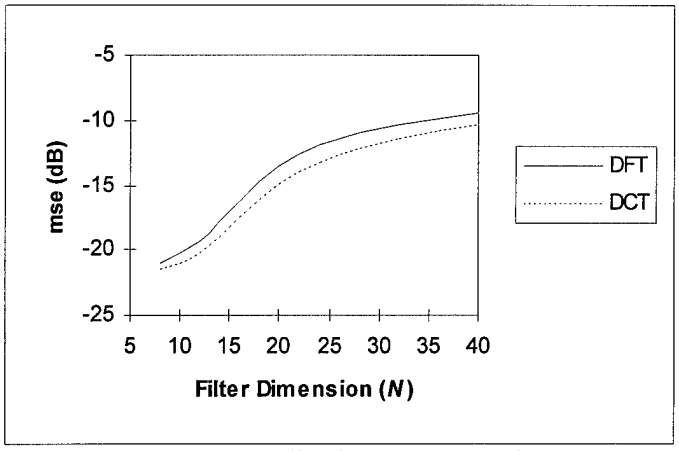



Figure 8. Normalized Mean-Squared Error vs. Filter Dimension

rate of $N/16$ pixels as $N$ increases. In each of the remaining Figures 9 - 11, the filter dimension, $N$, is held to $16 \times 16$ pixels. Figure 9 shows how the MSE decreases as $\rho$ increases for PSF $1/e$ widths of 1 and 8 pixels, with the SNR fixed at 20 dB. Figure 10 shows the MSE decreasing with increasing SNR for values of $\rho$ of 0.5, 0.9, and 0.99, with the PSF $1/e$ width fixed at 1 pixel.

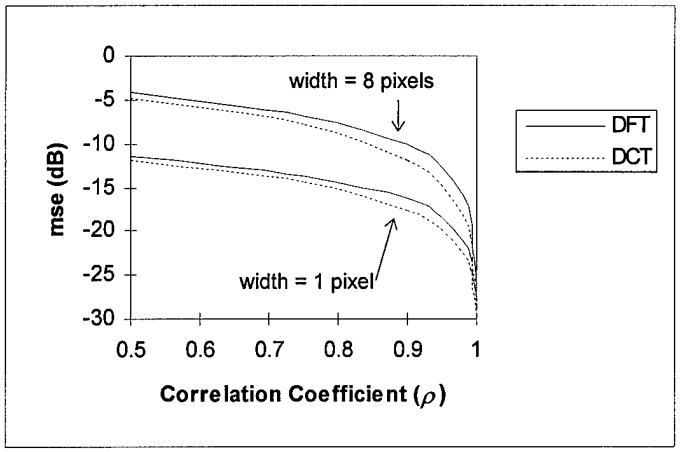



Figure 9. Normalized Mean-Squared Error
vs. Correlation Coefficient

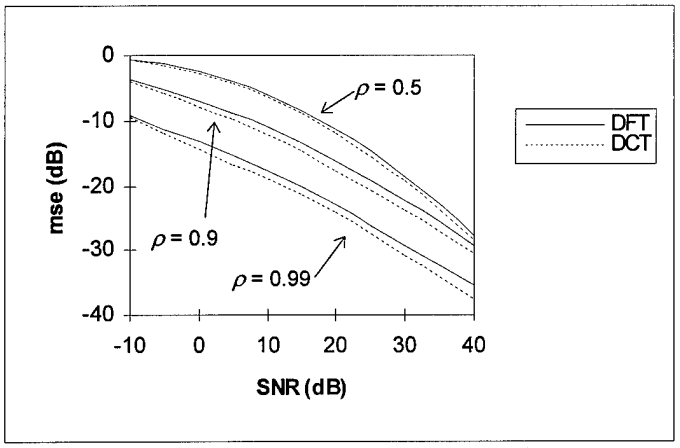



Figure 10. Normalized Mean-Squared Error
vs. Signal-to-Noise Ratio (SNR)

Finally Figure 11 shows the MSE increasing as the filter $1/e$ width increases and the filter band-width decreases. The figure shows curves for SNRs of 0, 20, and 40 dB, with $\rho$ fixed at 0.9.

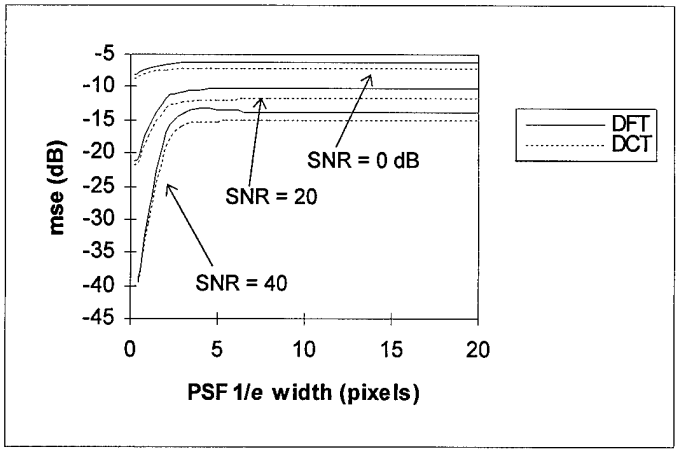



Figure 11. Normalized Mean-Squared Error vs. Point Spread Function (PSF) 1/*e* Width

In all of the cases tested, the trigonometric scalar Wiener filter performed better than the Fourier scalar Wiener filter, often demonstrating an improvement in mean-squared error on the order of 20 - 35%. The greatest increase in performance occurs as the bandwidth of the degradation filter in the linear model decreases. The larger gap between the DFT and DCT curves at the higher SNRs in Figure 11 demonstrate this effect. The fact that the gap is greater in Figure 9 for the case where the PSF 1/*e* width is 8 pixels than it is for the case where it is 1 pixel also demonstrates this improvement. The increased performance results from the fact that a type-II DCT possesses a near optimum energy compaction property about the low-frequency indices for highly correlated images [27]. The improvement with increasing correlation appears in Figures 9 and 10.

### 5.3 *Computational Complexity*

In addition to its improved MSE performance, the trigonometric scalar Wiener filter has the advantage of requiring fewer calculations to implement. The advantage to performing filtering in the trigonometric transform domain is that for real sequences, the transform domain coefficients are also all real, where they are complex in the Fourier domain. Fast algorithms exist for

the DCT based entirely on real arithmetic which operate with the same number of floating point operations as fast Fourier transforms [5], [27], [39]. These real-arithmetic algorithms for the DCT amount to a tremendous savings in computational performance since complex additions require twice the number of floating point operations as real additions and complex multiplications typically require six times the number of floating point operations as real multiplications. Fast hardware realizations of these trigonometric image processing techniques are also possible because of their computational cost savings.

In this chapter the performance of the scalar Wiener filter has been demonstrated visually through the use of an example, and algebraically through calculations of the filter's mean squared error performance. An argument has been presented that the filter also possesses the additional advantage of requiring fewer calculations to implement. Thus the filter designed during the course of this dissertation research not only has better performance than its Fourier equivalent, but it is easier to compute as well.

# VI. *Conclusions*

A summary of the major results and contributions of the research conducted for this dissertation is presented in this chapter. Some directions for future research in this area of study are also given.

## 6.1 *Results and Contributions*

The results of this research demonstrate how to apply the recently-developed symmetric convolution-multiplication property of the discrete trigonometric transforms [34] to the traditional image reconstruction problems of inverse and Wiener filtering. Each of the forty forms of the symmetric convolution-multiplication property for discrete trigonometric transforms is shown to exist as a vector-matrix operation. The convolutional forms of the trigonometric transforms can then diagonalize a matrix which represents the symmetric convolution operation [9], [11]. Derived in this manner, the symmetric convolution-multiplication property extends easily to multidimensional asymmetric sequences which represent the most general type of sequences encountered in practice.

The new diagonal forms for symmetric convolution matrices represent the first time such forms have been developed for the entire family of all forty cases of symmetric convolution. Although diagonal forms have been shown to exist for some of the forty cases [41], [42], and the initial discovery of a symmetric convolution-multiplication property for discrete trigonometric transforms [34] was not part of this research, the diagonal forms derived during the course of this research still make a significant contribution to the depth of knowledge surrounding symmetric convolution. As a result of this research, the relationship between the diagonal properties of dis-

crete Fourier transform matrices and discrete trigonometric transform matrices has been related for the first time.

The diagonalizing forms developed through this research permit a clearer understanding of the principle of how symmetric convolution in the sequence domain is equivalent to multiplication in the transform domain. With the new matrix formalism of symmetric convolution, it is much easier to visualize how the symmetric convolution-multiplication property extends to sequences in multiple dimensions and which might be asymmetric. The filtering of multidimensional asymmetric sequences is then possible because of the equivalence of symmetric convolution to multiplication in the transform domain for each of the underlying types of symmetry in an asymmetric image.

The research presented here uses the newly-derived vector-matrix form of symmetric convolution to calculate for the first time inverse and scalar Wiener filters in the trigonometric transform domain [10]. The new forms of the inverse and scalar Wiener filters closely resemble their traditional Fourier domain counterparts. Specifically, the new forms of scalar Wiener filters [12] are possible only because of the newly-developed symmetric convolution multiplication property of discrete trigonometric transforms [34]. Previous applications of the discrete cosine transform to linear filtering [26], [27] provided very good diagonal approximations for certain types of covariance matrices. The trigonometric transforms could not, however, simultaneously diagonalize a matrix representing degradation in the linear model. Now, with the symmetric convolution-multiplication property, very good scalar filter approximations are possible because an exact diagonalization of the degradation matrix is achievable, while still retaining the near optimum approximation to the diagonal form of the object covariance matrix.

The two-dimensional scalar Wiener filter is, however, limited by the type of symmetry present in the point spread function in certain cases. In general, a two-dimensional sequence can

possess four types of underlying symmetry. It can have antisymmetric and symmetric components about each of its two axes. The two-dimensional scalar Wiener filter derived during the course of this research is found to exist in the trigonometric transform domain only in cases where at most two of the four types of underlying symmetry are present at one time. This limitation is not severe because the two-dimensional version of the scalar Wiener filter which has whole-sample symmetry about both axes is claimed to be the most useful case for image processing. The whole-sample symmetric case is thus the focus for analyzing the performance of the scalar Wiener filter.

It was demonstrated that a scalar Wiener filter provides better mean-squared error performance for symmetric point spread functions while reducing the required number of computations. The trigonometric transform domain realizations of scalar Wiener filters use fewer calculations than traditional Fourier realizations of the same types of filters because they use entirely real arithmetic for real inputs. The fact that these filters provide better performance with fewer calculations should prove beneficial to a wide variety of other communications, signal processing, and control system applications, even though they were derived specifically for image reconstruction.

The need to improve the quality of noisy, blurred images of spaceborne objects is what initially prompted the Air Force to sponsor this design of two-dimensional trigonometric scalar Wiener filters. The large number of computations required for nonlinear, iterative image reconstruction techniques launched this reinvestigation of ways to improve traditional linear approaches. The results of this research show that the performance of traditional linear image reconstruction methods can be enhanced while also reducing the number of computations required. It is hoped that even better performance can be achieved by applying the symmetric convolution-

multiplication property of discrete trigonometric transforms to some of the other more powerful nonlinear image reconstruction techniques by reducing their number of calculations as well.

### 6.2 *Directions for Future Research*

Any future research which applies the symmetric convolution-multiplication property of discrete trigonometric transforms to image reconstruction should attempt to compensate for the assumptions which were required to conduct this research. This research required many assumptions because it was the first time symmetric convolution had ever been used in image reconstruction. Two assumptions which can be eliminated through future research are the assumption of an object-independent noise model and the assumption that the point spread function which distorts the object is completely known.

The imaging model for this research requires noise that is zero-mean, additive, of uniform variance, uncorrelated with itself, and independent of the object. For real-world imaging situations, this noise model is not as accurate as one which assumes that the noise is an object-dependent Poisson random process [17]. The filtering techniques derived here can be improved by using a more accurate Poisson noise model. Incorporating object-dependent photon noise into the imaging scenario would make the techniques developed here compatible with more modern image reconstruction methods.

The second assumption which can be eliminated through future research is that the trigonometric inverse and Wiener filters derived here require knowledge of the point spread function of the degrading system. In practice the point spread function of the degrading system is often not known exactly. This lack of knowledge leads to a technique called blind deconvolution [4], [45]. In blind deconvolution an estimate is sought for both the object and the unknown point spread function. In blind deconvolution, the point spread function is unknown but deterministic.

Two iterative techniques use inverse [2] and scalar Wiener filters [8] in the Fourier domain to estimate the object and the unknown point spread function. Both of these techniques should benefit from the increased performance of trigonometric inverse and scalar Wiener filters. The estimates should be of better quality and the iterations should converge to a solution faster. The number of computations should also be reduced because the trigonometric filters use real arithmetic. If future research reveals success with these iterative techniques, then the symmetric convolution-multiplication property could be applied to other more advanced techniques such as bispectrum processing for multiframe blind deconvolution [40].

A final area for future research which allows the point spread function to be unknown is to model the point spread function as a random process. Reasonable statistical approximations to a wide variety of atmospheric degradations can often be found [40]. These statistical approximations lead to wide variety of powerful techniques [13], [14] which require knowledge of the statistics of the distorting process but not the point spread function itself. These techniques all use Fourier domain representations which could benefit from being recast into the trigonometric transform domain.

Theoretically it can be concluded from the promising results of this research that almost any Fourier-domain image reconstruction technique should be able to benefit from being recast into the trigonometric transform domain.

## *Appendix: Derivation of the Two-Dimensional Scalar Wiener Filter for Trigonometric Transforms*

Many of the significant steps in the derivation of the two-dimensional scalar Wiener filter for trigonometric transforms are presented in this appendix. The end result is the $7 \times 4$ matrix equation displayed in Eq. (132) of Section 4.3.2. The derivation starts with Eq. (131) which is repeated below

$$\begin{aligned}\left(\boldsymbol{F}_{aa}+\boldsymbol{F}_{as}+\boldsymbol{F}_{sa}+\boldsymbol{F}_{ss}\right)\left[\left(\boldsymbol{H}_{aa}+\boldsymbol{H}_{as}+\boldsymbol{H}_{sa}+\boldsymbol{H}_{ss}\right)\boldsymbol{R}_{\theta\theta}\left(\boldsymbol{H}_{aa}+\boldsymbol{H}_{as}+\boldsymbol{H}_{sa}+\boldsymbol{H}_{ss}\right)^{T}+\boldsymbol{R}_{ww}\right] \\ =\boldsymbol{R}_{\theta\theta}\left(\boldsymbol{H}_{aa}+\boldsymbol{H}_{as}+\boldsymbol{H}_{aa}+\boldsymbol{H}_{as}\right)^{T}.\end{aligned} \tag{131}$$

Recall that Eq. (131) is the solution resulting from the Bayesian Gauss-Markov Theorem [28] for two-dimensional asymmetric filters represented by the matrices $\boldsymbol{F}=\boldsymbol{F}_{aa}+\boldsymbol{F}_{as}+\boldsymbol{F}_{sa}+\boldsymbol{F}_{ss}$ and $\boldsymbol{H}=\boldsymbol{H}_{aa}+\boldsymbol{H}_{as}+\boldsymbol{H}_{sa}+\boldsymbol{H}_{ss}$. The matrix $\boldsymbol{H}$ performs symmetric convolution to implement the point spread function (PSF) of a system which degrades an object represented by the vector $\theta$. The matrix $\boldsymbol{F}$ performs symmetric convolution to recover a vector estimate, $\hat{\theta}$, of the original object.

Expanding the terms on the left-hand side of Eq. (131) produces

$$\begin{aligned}&\boldsymbol{F}_{aa}\boldsymbol{H}_{aa}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{aa}^{T}+\boldsymbol{F}_{aa}\boldsymbol{H}_{aa}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{as}^{T}+\boldsymbol{F}_{aa}\boldsymbol{H}_{aa}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{sa}^{T}+\boldsymbol{F}_{aa}\boldsymbol{H}_{aa}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{ss}^{T}\\&+\boldsymbol{F}_{aa}\boldsymbol{H}_{as}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{aa}^{T}+\boldsymbol{F}_{aa}\boldsymbol{H}_{as}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{as}^{T}+\boldsymbol{F}_{aa}\boldsymbol{H}_{as}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{sa}^{T}+\boldsymbol{F}_{aa}\boldsymbol{H}_{as}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{ss}^{T}\\&+\boldsymbol{F}_{aa}\boldsymbol{H}_{sa}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{aa}^{T}+\boldsymbol{F}_{aa}\boldsymbol{H}_{sa}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{as}^{T}+\boldsymbol{F}_{aa}\boldsymbol{H}_{sa}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{sa}^{T}+\boldsymbol{F}_{aa}\boldsymbol{H}_{sa}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{ss}^{T}\\&+\boldsymbol{F}_{aa}\boldsymbol{H}_{ss}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{aa}^{T}+\boldsymbol{F}_{aa}\boldsymbol{H}_{ss}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{as}^{T}+\boldsymbol{F}_{aa}\boldsymbol{H}_{ss}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{sa}^{T}+\boldsymbol{F}_{aa}\boldsymbol{H}_{ss}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{ss}^{T}\\&+\boldsymbol{F}_{as}\boldsymbol{H}_{aa}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{aa}^{T}+\boldsymbol{F}_{as}\boldsymbol{H}_{aa}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{as}^{T}+\boldsymbol{F}_{as}\boldsymbol{H}_{aa}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{sa}^{T}+\boldsymbol{F}_{as}\boldsymbol{H}_{aa}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{ss}^{T}\\&+\boldsymbol{F}_{as}\boldsymbol{H}_{as}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{aa}^{T}+\boldsymbol{F}_{as}\boldsymbol{H}_{as}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{as}^{T}+\boldsymbol{F}_{as}\boldsymbol{H}_{as}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{sa}^{T}+\boldsymbol{F}_{as}\boldsymbol{H}_{as}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{ss}^{T}\\&+\boldsymbol{F}_{as}\boldsymbol{H}_{sa}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{aa}^{T}+\boldsymbol{F}_{as}\boldsymbol{H}_{sa}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{as}^{T}+\boldsymbol{F}_{as}\boldsymbol{H}_{sa}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{sa}^{T}+\boldsymbol{F}_{as}\boldsymbol{H}_{sa}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{ss}^{T}\\&+\boldsymbol{F}_{as}\boldsymbol{H}_{ss}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{aa}^{T}+\boldsymbol{F}_{as}\boldsymbol{H}_{ss}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{as}^{T}+\boldsymbol{F}_{as}\boldsymbol{H}_{ss}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{sa}^{T}+\boldsymbol{F}_{as}\boldsymbol{H}_{ss}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{ss}^{T}\\&+\boldsymbol{F}_{sa}\boldsymbol{H}_{aa}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{aa}^{T}+\boldsymbol{F}_{sa}\boldsymbol{H}_{aa}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{as}^{T}+\boldsymbol{F}_{sa}\boldsymbol{H}_{aa}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{sa}^{T}+\boldsymbol{F}_{sa}\boldsymbol{H}_{aa}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{ss}^{T}\\&+\boldsymbol{F}_{sa}\boldsymbol{H}_{as}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{aa}^{T}+\boldsymbol{F}_{sa}\boldsymbol{H}_{as}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{as}^{T}+\boldsymbol{F}_{sa}\boldsymbol{H}_{as}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{sa}^{T}+\boldsymbol{F}_{sa}\boldsymbol{H}_{as}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{ss}^{T}\ \ldots\end{aligned}$$

$$
\begin{aligned}
&+\boldsymbol{F}_{sa}\boldsymbol{H}_{sa}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{aa}^{T}+\boldsymbol{F}_{sa}\boldsymbol{H}_{sa}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{as}^{T}+\boldsymbol{F}_{sa}\boldsymbol{H}_{sa}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{sa}^{T}+\boldsymbol{F}_{sa}\boldsymbol{H}_{sa}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{ss}^{T}\\
&+\boldsymbol{F}_{sa}\boldsymbol{H}_{ss}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{aa}^{T}+\boldsymbol{F}_{sa}\boldsymbol{H}_{ss}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{as}^{T}+\boldsymbol{F}_{sa}\boldsymbol{H}_{ss}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{sa}^{T}+\boldsymbol{F}_{sa}\boldsymbol{H}_{ss}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{ss}^{T}\\
&+\boldsymbol{F}_{ss}\boldsymbol{H}_{aa}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{aa}^{T}+\boldsymbol{F}_{ss}\boldsymbol{H}_{aa}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{as}^{T}+\boldsymbol{F}_{ss}\boldsymbol{H}_{aa}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{sa}^{T}+\boldsymbol{F}_{ss}\boldsymbol{H}_{aa}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{ss}^{T}\\
&+\boldsymbol{F}_{ss}\boldsymbol{H}_{as}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{aa}^{T}+\boldsymbol{F}_{ss}\boldsymbol{H}_{as}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{as}^{T}+\boldsymbol{F}_{ss}\boldsymbol{H}_{as}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{sa}^{T}+\boldsymbol{F}_{ss}\boldsymbol{H}_{as}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{ss}^{T}\\
&+\boldsymbol{F}_{ss}\boldsymbol{H}_{sa}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{aa}^{T}+\boldsymbol{F}_{ss}\boldsymbol{H}_{sa}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{as}^{T}+\boldsymbol{F}_{as}\boldsymbol{H}_{sa}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{sa}^{T}+\boldsymbol{F}_{ss}\boldsymbol{H}_{sa}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{ss}^{T}\\
&+\boldsymbol{F}_{ss}\boldsymbol{H}_{ss}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{aa}^{T}+\boldsymbol{F}_{ss}\boldsymbol{H}_{ss}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{as}^{T}+\boldsymbol{F}_{ss}\boldsymbol{H}_{ss}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{sa}^{T}+\boldsymbol{F}_{ss}\boldsymbol{H}_{ss}\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{ss}^{T}\\
&+\boldsymbol{F}_{aa}\boldsymbol{R}_{ww}+\boldsymbol{F}_{as}\boldsymbol{R}_{ww}+\boldsymbol{F}_{sa}\boldsymbol{R}_{ww}+\boldsymbol{F}_{ss}\boldsymbol{R}_{ww}\\
&=\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{aa}^{T}+\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{as}^{T}+\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{sa}^{T}+\boldsymbol{R}_{\theta\theta}\boldsymbol{H}_{ss}^{T}.
\end{aligned}
\tag{155}
$$

After substituting the trigonometric transform domain versions of the symmetric convolution matrices from Eqs. (76) - (91), Eq. (155) becomes

$$
\begin{aligned}
&\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{F}_{aa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^{T}\\
&+\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{F}_{aa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^{T}\\
&+\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{F}_{aa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^{T}\\
&+\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{F}_{aa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^{T}\\
&-\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{F}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^{T}\\
&-\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{F}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^{T}\\
&-\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{F}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^{T}\\
&-\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{F}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^{T}\cdots
\end{aligned}
$$

$$
\begin{aligned}
&-\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{F}_{aa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&-\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{F}_{aa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&-\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{F}_{aa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&-\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{F}_{aa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{F}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{F}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{F}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{F}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&-\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{F}_{as}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&-\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{F}_{as}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&-\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{F}_{as}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&-\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{F}_{as}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\cdots
\end{aligned}
$$

$$
\begin{aligned}
&-\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{F}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&-\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{F}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&-\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{F}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&-\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{F}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{F}_{as}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{F}_{as}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{F}_{as}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{F}_{as}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{F}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{F}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{F}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{F}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\cdots
\end{aligned}
$$

$$
\begin{aligned}
&-\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{F}_{sa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&-\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{F}_{sa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&-\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{F}_{sa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&-\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{F}_{sa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{F}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{F}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{F}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{F}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&-\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{F}_{sa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&-\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{F}_{sa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&-\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{F}_{sa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&-\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{F}_{sa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\cdots
\end{aligned}
$$

$$
\begin{aligned}
&+\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{F}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{F}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{F}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{F}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{F}_{ss}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{F}_{ss}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{F}_{ss}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{F}_{ss}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{F}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{F}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{F}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{F}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\cdots
\end{aligned}
$$

$$
\begin{aligned}
&+\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{F}_{ss}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{F}_{ss}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{F}_{ss}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{F}_{ss}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{F}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{F}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{F}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{F}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\\
&\qquad\cdot\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{F}_{aa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)\right]\boldsymbol{R}_{ww}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^T\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-T}\\
&-\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{F}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)\right]\boldsymbol{R}_{ww}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^T\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-T}\\
&-\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{F}_{sa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\boldsymbol{R}_{ww}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^T\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-T}\\
&+\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{F}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]\boldsymbol{R}_{ww}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^T\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-T}\\
&=\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-1}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{aa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-1}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{S}_{2e,N_2}^{-1}\right)\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-1}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{S}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{sa}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T\\
&+\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-1}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\boldsymbol{R}_{\theta\theta}\left[\left(\boldsymbol{C}_{2e,N_1}^{-1}\otimes\boldsymbol{C}_{2e,N_2}^{-1}\right)\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)\right]^T. \qquad (156)
\end{aligned}
$$

Equation (156) can be simplified by making the substitutions

$$\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} = \left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)\boldsymbol{R}_{\theta\theta}\left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^T, \tag{157}$$

$$\boldsymbol{R}_{w_{aa}w_{aa}} = \left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right)\boldsymbol{R}_{ww}\left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right)^T, \tag{158}$$

$$\boldsymbol{R}_{w_{as}w_{as}} = \left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right)\boldsymbol{R}_{ww}\left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right)^T, \tag{159}$$

$$\boldsymbol{R}_{w_{sa}w_{sa}} = \left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)\boldsymbol{R}_{ww}\left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^T, \tag{160}$$

and

$$\boldsymbol{R}_{w_{ss}w_{ss}} = \left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)\boldsymbol{R}_{ww}\left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^T. \tag{161}$$

Making the substitutions from Eqs. (157) - (161) into Eq. (156) produces

$$\begin{aligned}
&\left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^{-1}\mathcal{F}_{aa}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}\left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right)^{-T} \\
&+\left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^{-1}\mathcal{F}_{aa}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right)^{-T} \\
&+\left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^{-1}\mathcal{F}_{aa}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}\left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^{-T} \\
&+\left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^{-1}\mathcal{F}_{aa}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^{-T} \\
&-\left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^{-1}\mathcal{F}_{aa}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}\left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right)^{-T} \\
&-\left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^{-1}\mathcal{F}_{aa}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right)^{-T} \\
&-\left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^{-1}\mathcal{F}_{aa}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}\left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^{-T} \\
&-\left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^{-1}\mathcal{F}_{aa}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^{-T} \\
&-\left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right)^{-1}\mathcal{F}_{aa}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}\left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right)^{-T} \\
&-\left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right)^{-1}\mathcal{F}_{aa}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right)^{-T} \\
&-\left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right)^{-1}\mathcal{F}_{aa}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}\left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^{-T} \\
&-\left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right)^{-1}\mathcal{F}_{aa}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^{-T} \\
&+\left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right)^{-1}\mathcal{F}_{aa}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}\left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right)^{-T} \\
&+\left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right)^{-1}\mathcal{F}_{aa}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right)^{-T} \\
&+\left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right)^{-1}\mathcal{F}_{aa}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}\left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^{-T} \\
&+\left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right)^{-1}\mathcal{F}_{aa}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^{-T} \ldots
\end{aligned}$$

$$
\begin{aligned}
&-\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-1}\mathcal{F}_{as}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-T}\\
&-\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-1}\mathcal{F}_{as}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-T}\\
&-\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-1}\mathcal{F}_{as}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-T}\\
&-\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-1}\mathcal{F}_{as}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-T}\\
&-\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-1}\mathcal{F}_{as}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-T}\\
&-\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-1}\mathcal{F}_{as}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-T}\\
&-\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-1}\mathcal{F}_{as}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-T}\\
&-\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-1}\mathcal{F}_{as}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-1}\mathcal{F}_{as}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-1}\mathcal{F}_{as}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-1}\mathcal{F}_{as}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-1}\mathcal{F}_{as}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-1}\mathcal{F}_{as}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-1}\mathcal{F}_{as}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-1}\mathcal{F}_{as}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-1}\mathcal{F}_{as}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-T}\\
&-\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-1}\mathcal{F}_{sa}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-T}\\
&-\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-1}\mathcal{F}_{sa}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-T}\\
&-\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-1}\mathcal{F}_{sa}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-T}\\
&-\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-1}\mathcal{F}_{sa}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-1}\mathcal{F}_{sa}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-1}\mathcal{F}_{sa}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-1}\mathcal{F}_{sa}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-1}\mathcal{F}_{sa}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-T}\\
&-\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-1}\mathcal{F}_{sa}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-T}\\
&-\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-1}\mathcal{F}_{sa}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-T}\ldots
\end{aligned}
$$

$$
\begin{aligned}
&-\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-1}\mathcal{F}_{sa}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-T}\\
&-\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-1}\mathcal{F}_{sa}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-1}\mathcal{F}_{sa}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-1}\mathcal{F}_{sa}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-1}\mathcal{F}_{sa}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-1}\mathcal{F}_{sa}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-1}\mathcal{F}_{ss}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-1}\mathcal{F}_{ss}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-1}\mathcal{F}_{ss}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-1}\mathcal{F}_{ss}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-1}\mathcal{F}_{ss}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-1}\mathcal{F}_{ss}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-1}\mathcal{F}_{ss}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-1}\mathcal{F}_{ss}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-1}\mathcal{F}_{ss}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-1}\mathcal{F}_{ss}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-1}\mathcal{F}_{ss}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-1}\mathcal{F}_{ss}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-1}\mathcal{F}_{ss}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-1}\mathcal{F}_{ss}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-1}\mathcal{F}_{ss}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-1}\mathcal{F}_{ss}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-1}\mathcal{F}_{aa}\boldsymbol{R}_{\mathcal{W}_{aa}\mathcal{W}_{aa}}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-T}\\
&-\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-1}\mathcal{F}_{as}\boldsymbol{R}_{\mathcal{W}_{as}\mathcal{W}_{as}}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-T}\\
&-\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-1}\mathcal{F}_{sa}\boldsymbol{R}_{\mathcal{W}_{sa}\mathcal{W}_{sa}}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-1}\mathcal{F}_{ss}\boldsymbol{R}_{\mathcal{W}_{ss}\mathcal{W}_{ss}}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-T}\cdots
\end{aligned}
$$

$$
\begin{aligned}
&= \left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^{-1} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{aa} \left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right)^{-T} \\
&+ \left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^{-1} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{as} \left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right)^{-T} \\
&+ \left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^{-1} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{sa} \left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^{-T} \\
&+ \left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^{-1} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{ss} \left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^{-T}. \qquad (162)
\end{aligned}
$$

The simplification of Eq. (156) to produce Eq. (162) makes use of the facts that $\mathcal{H}_{aa} = \mathcal{H}_{aa}^T$, $\mathcal{H}_{as} = \mathcal{H}_{as}^T$, $\mathcal{H}_{sa} = \mathcal{H}_{sa}^T$, and $\mathcal{H}_{ss} = \mathcal{H}_{ss}^T$ because the matrices $\mathcal{H}_{aa}$, $\mathcal{H}_{as}$, $\mathcal{H}_{sa}$, and $\mathcal{H}_{ss}$ are all diagonal. Combining terms in Eq. (162) results in the expression

$$
\begin{aligned}
&\left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^{-1} \Big[ \mathcal{F}_{aa} \mathcal{H}_{aa} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{aa} - \mathcal{F}_{as} \mathcal{H}_{as} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{aa} + \mathcal{F}_{aa} \boldsymbol{R}_{w_{aa}w_{aa}} \\
&\qquad - \mathcal{F}_{sa} \mathcal{H}_{sa} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{aa} + \mathcal{F}_{ss} \mathcal{H}_{ss} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{aa} \Big] \left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right)^{-T} \\
&+ \left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^{-1} \Big[ \mathcal{F}_{aa} \mathcal{H}_{aa} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{as} - \mathcal{F}_{as} \mathcal{H}_{as} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{as} - \mathcal{F}_{as} \boldsymbol{R}_{w_{as}w_{as}} \\
&\qquad - \mathcal{F}_{sa} \mathcal{H}_{sa} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{as} + \mathcal{F}_{ss} \mathcal{H}_{ss} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{as} \Big] \left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right)^{-T} \\
&+ \left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^{-1} \Big[ \mathcal{F}_{aa} \mathcal{H}_{aa} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{sa} - \mathcal{F}_{as} \mathcal{H}_{as} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{sa} - \mathcal{F}_{sa} \boldsymbol{R}_{w_{sa}w_{sa}} \\
&\qquad - \mathcal{F}_{sa} \mathcal{H}_{sa} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{sa} + \mathcal{F}_{ss} \mathcal{H}_{ss} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{sa} \Big] \left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^{-T} \\
&+ \left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^{-1} \Big[ \mathcal{F}_{aa} \mathcal{H}_{aa} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{ss} - \mathcal{F}_{as} \mathcal{H}_{as} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{ss} + \mathcal{F}_{ss} \boldsymbol{R}_{w_{ss}w_{ss}} \\
&\qquad - \mathcal{F}_{sa} \mathcal{H}_{sa} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{ss} + \mathcal{F}_{ss} \mathcal{H}_{ss} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{ss} \Big] \left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^{-T} \\
&+ \left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^{-1} \Big[ -\mathcal{F}_{aa} \mathcal{H}_{as} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{aa} - \mathcal{F}_{as} \mathcal{H}_{aa} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{aa} \\
&\qquad + \mathcal{F}_{sa} \mathcal{H}_{ss} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{aa} + \mathcal{F}_{ss} \mathcal{H}_{sa} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{aa} \Big] \left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right)^{-T} \\
&+ \left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^{-1} \Big[ -\mathcal{F}_{aa} \mathcal{H}_{as} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{as} - \mathcal{F}_{as} \mathcal{H}_{aa} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{as} \\
&\qquad + \mathcal{F}_{sa} \mathcal{H}_{ss} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{as} + \mathcal{F}_{ss} \mathcal{H}_{sa} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{as} \Big] \left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right)^{-T} \\
&+ \left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^{-1} \Big[ -\mathcal{F}_{aa} \mathcal{H}_{as} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{sa} - \mathcal{F}_{as} \mathcal{H}_{aa} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{sa} \\
&\qquad + \mathcal{F}_{sa} \mathcal{H}_{ss} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{sa} + \mathcal{F}_{ss} \mathcal{H}_{sa} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{sa} \Big] \left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^{-T} \\
&+ \left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^{-1} \Big[ -\mathcal{F}_{aa} \mathcal{H}_{as} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{ss} - \mathcal{F}_{as} \mathcal{H}_{aa} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{ss} \\
&\qquad + \mathcal{F}_{sa} \mathcal{H}_{ss} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{ss} + \mathcal{F}_{ss} \mathcal{H}_{sa} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{ss} \Big] \left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{C}_{2e,N_2}\right)^{-T} \\
&+ \left(\boldsymbol{C}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right)^{-1} \Big[ -\mathcal{F}_{aa} \mathcal{H}_{sa} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{aa} + \mathcal{F}_{as} \mathcal{H}_{ss} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{aa} \\
&\qquad - \mathcal{F}_{sa} \mathcal{H}_{aa} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{aa} + \mathcal{F}_{ss} \mathcal{H}_{as} \boldsymbol{R}_{\Theta_{ss}\Theta_{ss}} \mathcal{H}_{aa} \Big] \left(\boldsymbol{S}_{2e,N_1} \otimes \boldsymbol{S}_{2e,N_2}\right)^{-T} \cdots
\end{aligned}
$$

$$
\begin{aligned}
&+\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-1}\left[-\mathcal{F}_{aa}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}+\mathcal{F}_{as}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}\right.\\
&\qquad\left.-\mathcal{F}_{sa}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}+\mathcal{F}_{ss}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}\right]\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-1}\left[-\mathcal{F}_{aa}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}+\mathcal{F}_{as}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}\right.\\
&\qquad\left.-\mathcal{F}_{sa}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}+\mathcal{F}_{ss}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}\right]\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-1}\left[-\mathcal{F}_{aa}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}+\mathcal{F}_{as}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}\right.\\
&\qquad\left.-\mathcal{F}_{sa}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}+\mathcal{F}_{ss}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}\right]\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-1}\left[\mathcal{F}_{aa}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}+\mathcal{F}_{as}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}\right.\\
&\qquad\left.+\mathcal{F}_{sa}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}+\mathcal{F}_{ss}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}\right]\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-1}\left[\mathcal{F}_{aa}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}+\mathcal{F}_{as}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}\right.\\
&\qquad\left.+\mathcal{F}_{sa}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}+\mathcal{F}_{ss}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}\right]\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-1}\left[\mathcal{F}_{aa}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}+\mathcal{F}_{as}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}\right.\\
&\qquad\left.+\mathcal{F}_{sa}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}+\mathcal{F}_{ss}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}\right]\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-1}\left[\mathcal{F}_{aa}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}+\mathcal{F}_{as}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}\right.\\
&\qquad\left.+\mathcal{F}_{sa}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}+\mathcal{F}_{ss}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}\right]\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-T}\\
&=\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-1}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-1}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{S}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-1}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}\left(\boldsymbol{S}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-T}\\
&+\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-1}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}\left(\boldsymbol{C}_{2e,N_1}\otimes\boldsymbol{C}_{2e,N_2}\right)^{-T}. \qquad (163)
\end{aligned}
$$

Equality in Eq. (163) will hold only if

$$
\begin{aligned}
&\mathcal{F}_{aa}\left(\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}+\boldsymbol{R}_{w_{aa}w_{aa}}\right)-\mathcal{F}_{as}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}-\mathcal{F}_{sa}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}\\
&\qquad+\mathcal{F}_{ss}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}=\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}, \qquad (164)
\end{aligned}
$$

$$
\begin{aligned}
&\mathcal{F}_{aa}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}-\mathcal{F}_{as}\left(\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}+\boldsymbol{R}_{w_{as}w_{as}}\right)-\mathcal{F}_{sa}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}\\
&\qquad+\mathcal{F}_{ss}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}=\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}, \qquad (165)
\end{aligned}
$$

$$
\begin{aligned}
&\mathcal{F}_{aa}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}-\mathcal{F}_{as}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}-\mathcal{F}_{sa}\left(\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}+\boldsymbol{R}_{w_{sa}w_{sa}}\right)\\
&\qquad+\mathcal{F}_{ss}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}=\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}, \qquad (166)
\end{aligned}
$$

$$\mathcal{F}_{aa}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}-\mathcal{F}_{as}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}-\mathcal{F}_{sa}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}$$
$$+\mathcal{F}_{ss}\left(\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}+\boldsymbol{R}_{w_{ss}w_{ss}}\right)=\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}, \tag{167}$$

$$-\mathcal{F}_{aa}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}-\mathcal{F}_{as}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}+\mathcal{F}_{sa}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}$$
$$+\mathcal{F}_{ss}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}=\boldsymbol{0}, \tag{168}$$

$$-\mathcal{F}_{aa}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}-\mathcal{F}_{as}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}+\mathcal{F}_{sa}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}$$
$$+\mathcal{F}_{ss}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}=\boldsymbol{0}, \tag{169}$$

$$-\mathcal{F}_{aa}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}-\mathcal{F}_{as}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}+\mathcal{F}_{sa}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}$$
$$+\mathcal{F}_{ss}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}=\boldsymbol{0}, \tag{170}$$

$$-\mathcal{F}_{aa}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}-\mathcal{F}_{as}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}+\mathcal{F}_{sa}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}$$
$$+\mathcal{F}_{ss}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}=\boldsymbol{0}, \tag{171}$$

$$-\mathcal{F}_{aa}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}+\mathcal{F}_{as}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}-\mathcal{F}_{sa}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}$$
$$+\mathcal{F}_{ss}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}=\boldsymbol{0}, \tag{172}$$

$$-\mathcal{F}_{aa}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}+\mathcal{F}_{as}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}-\mathcal{F}_{sa}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}$$
$$+\mathcal{F}_{ss}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}=\boldsymbol{0}, \tag{173}$$

$$-\mathcal{F}_{aa}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}+\mathcal{F}_{as}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}-\mathcal{F}_{sa}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}$$
$$+\mathcal{F}_{ss}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}=\boldsymbol{0}, \tag{174}$$

$$-\mathcal{F}_{aa}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}+\mathcal{F}_{as}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}-\mathcal{F}_{sa}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}$$
$$+\mathcal{F}_{ss}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss}=\boldsymbol{0}, \tag{175}$$

$$\mathcal{F}_{aa}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}+\mathcal{F}_{as}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}+\mathcal{F}_{sa}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}$$
$$+\mathcal{F}_{ss}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{aa}=\boldsymbol{0}, \tag{176}$$

$$\mathcal{F}_{aa}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}+\mathcal{F}_{as}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}+\mathcal{F}_{sa}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}$$
$$+\mathcal{F}_{ss}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{as}=\boldsymbol{0}, \tag{177}$$

$$\mathcal{F}_{aa}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}+\mathcal{F}_{as}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}+\mathcal{F}_{sa}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}$$
$$+\mathcal{F}_{ss}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{sa}=\boldsymbol{0}, \tag{178}$$

and $$\mathcal{F}_{aa}\mathcal{H}_{ss}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss} + \mathcal{F}_{as}\mathcal{H}_{sa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss} + \mathcal{F}_{sa}\mathcal{H}_{as}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss} + \mathcal{F}_{ss}\mathcal{H}_{aa}\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}\mathcal{H}_{ss} = \mathbf{0}. \quad (179)$$

Just as in the one-dimensional derivation of the scalar Wiener filter, every matrix in Eqs. (164) - (179) is either diagonal or approximately diagonal. The matrices $\mathcal{F}_{aa}$, $\mathcal{F}_{as}$, $\mathcal{F}_{sa}$, $\mathcal{F}_{ss}$, $\mathcal{H}_{aa}$, $\mathcal{H}_{as}$, $\mathcal{H}_{sa}$, and $\mathcal{H}_{ss}$ will always be diagonal. The transform domain noise correlation matrices, $\boldsymbol{R}_{\mathcal{W}_{aa}\mathcal{W}_{aa}}$, $\boldsymbol{R}_{\mathcal{W}_{as}\mathcal{W}_{as}}$, $\boldsymbol{R}_{\mathcal{W}_{sa}\mathcal{W}_{sa}}$, and $\boldsymbol{R}_{\mathcal{W}_{ss}\mathcal{W}_{ss}}$, will be diagonal because the noise samples are uncorrelated and have uniform variance. The matrices $\boldsymbol{R}_{\mathcal{W}_{aa}\mathcal{W}_{aa}}$, $\boldsymbol{R}_{\mathcal{W}_{as}\mathcal{W}_{as}}$, $\boldsymbol{R}_{\mathcal{W}_{sa}\mathcal{W}_{sa}}$, and $\boldsymbol{R}_{\mathcal{W}_{ss}\mathcal{W}_{ss}}$ are thus exactly represented by their diagonal elements $\overline{\mathcal{W}_{aa}^2(k_1,k_2)}$, $\overline{\mathcal{W}_{as}^2(k_1,k_2)}$, $\overline{\mathcal{W}_{sa}^2(k_1,k_2)}$, and $\overline{\mathcal{W}_{ss}^2(k_1,k_2)}$. The transform-domain object correlation matrix, $\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}$, will be well-approximated by its diagonal elements based on the same assumption of a highly-correlated object used for the one-dimensional case. The matrix $\boldsymbol{R}_{\Theta_{ss}\Theta_{ss}}$ is therefore well-approximated by its diagonal elements $\overline{\Theta_{ss}^2(k_1,k_2)}$.

The above approximations allow for a scalar solution based solely on the diagonal elements of the matrices in Eqs. (164) - (179). Solving for the diagonal elements of Eqs. (164) - (179) produces the $16\times 4$ overdetermined matrix equation which appears as Eq. (180) on the following page. Each entry in Eq. (180) is dependent on the transform domain index variables $k_1$ and $k_2$, but this dependence is suppressed in the equation to conserve space. The $16\times 4$ matrix equation in Eq. (180) reduces to the $7\times 4$ matrix equation of Eq. (132) from Section 4.3.2 which is repeated here following Eq. (180).

$$
\begin{bmatrix}
\mathcal{H}_{aa}^2 + \frac{\overline{\mathcal{W}_{aa}^2}}{\Theta_{ss}^2} & -\mathcal{H}_{aa}\mathcal{H}_{as} & -\mathcal{H}_{aa}\mathcal{H}_{sa} & \mathcal{H}_{aa}\mathcal{H}_{ss} \\
-\mathcal{H}_{aa}\mathcal{H}_{as} & \mathcal{H}_{as}^2 + \frac{\overline{\mathcal{W}_{as}^2}}{\Theta_{ss}^2} & \mathcal{H}_{as}\mathcal{H}_{sa} & -\mathcal{H}_{as}\mathcal{H}_{ss} \\
-\mathcal{H}_{aa}\mathcal{H}_{sa} & \mathcal{H}_{as}\mathcal{H}_{sa} & \mathcal{H}_{sa}^2 + \frac{\overline{\mathcal{W}_{sa}^2}}{\Theta_{ss}^2} & -\mathcal{H}_{sa}\mathcal{H}_{ss} \\
\mathcal{H}_{aa}\mathcal{H}_{ss} & -\mathcal{H}_{as}\mathcal{H}_{ss} & -\mathcal{H}_{sa}\mathcal{H}_{ss} & \mathcal{H}_{ss}^2 + \frac{\overline{\mathcal{W}_{ss}^2}}{\Theta_{ss}^2} \\
\mathcal{H}_{aa}\mathcal{H}_{as} & \mathcal{H}_{aa}^2 & -\mathcal{H}_{aa}\mathcal{H}_{ss} & -\mathcal{H}_{aa}\mathcal{H}_{sa} \\
\mathcal{H}_{as}^2 & \mathcal{H}_{aa}\mathcal{H}_{as} & -\mathcal{H}_{as}\mathcal{H}_{ss} & -\mathcal{H}_{as}\mathcal{H}_{sa} \\
-\mathcal{H}_{as}\mathcal{H}_{sa} & -\mathcal{H}_{aa}\mathcal{H}_{sa} & \mathcal{H}_{sa}\mathcal{H}_{ss} & \mathcal{H}_{sa}^2 \\
-\mathcal{H}_{as}\mathcal{H}_{ss} & -\mathcal{H}_{aa}\mathcal{H}_{ss} & \mathcal{H}_{ss}^2 & \mathcal{H}_{sa}\mathcal{H}_{ss} \\
\mathcal{H}_{aa}\mathcal{H}_{sa} & -\mathcal{H}_{aa}\mathcal{H}_{ss} & \mathcal{H}_{aa}^2 & -\mathcal{H}_{aa}\mathcal{H}_{as} \\
-\mathcal{H}_{as}\mathcal{H}_{sa} & \mathcal{H}_{as}\mathcal{H}_{ss} & -\mathcal{H}_{aa}\mathcal{H}_{as} & \mathcal{H}_{as}^2 \\
\mathcal{H}_{sa}^2 & -\mathcal{H}_{sa}\mathcal{H}_{ss} & \mathcal{H}_{aa}\mathcal{H}_{sa} & -\mathcal{H}_{as}\mathcal{H}_{sa} \\
-\mathcal{H}_{sa}\mathcal{H}_{ss} & \mathcal{H}_{ss}^2 & -\mathcal{H}_{aa}\mathcal{H}_{ss} & \mathcal{H}_{as}\mathcal{H}_{ss} \\
\mathcal{H}_{aa}\mathcal{H}_{ss} & \mathcal{H}_{aa}\mathcal{H}_{sa} & \mathcal{H}_{aa}\mathcal{H}_{as} & \mathcal{H}_{aa}^2 \\
\mathcal{H}_{as}\mathcal{H}_{ss} & \mathcal{H}_{as}\mathcal{H}_{sa} & \mathcal{H}_{as}^2 & \mathcal{H}_{aa}\mathcal{H}_{as} \\
\mathcal{H}_{sa}\mathcal{H}_{ss} & \mathcal{H}_{sa}^2 & \mathcal{H}_{as}\mathcal{H}_{sa} & \mathcal{H}_{aa}\mathcal{H}_{sa} \\
\mathcal{H}_{ss}^2 & \mathcal{H}_{sa}\mathcal{H}_{ss} & \mathcal{H}_{as}\mathcal{H}_{ss} & \mathcal{H}_{aa}\mathcal{H}_{ss}
\end{bmatrix}
\begin{bmatrix}
\mathcal{F}_{aa} \\ \mathcal{F}_{as} \\ \mathcal{F}_{sa} \\ \mathcal{F}_{ss}
\end{bmatrix}
=
\begin{bmatrix}
\mathcal{H}_{aa} \\ -\mathcal{H}_{as} \\ -\mathcal{H}_{sa} \\ \mathcal{H}_{ss} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0
\end{bmatrix}
\tag{180}
$$

$$
\begin{bmatrix}
\mathcal{H}_{aa}^2 + \dfrac{\overline{\mathcal{W}_{aa}^2}}{\overline{\Theta_{ss}^2}} & -\mathcal{H}_{aa}\mathcal{H}_{as} & -\mathcal{H}_{aa}\mathcal{H}_{sa} & \mathcal{H}_{aa}\mathcal{H}_{ss} \\
-\mathcal{H}_{aa}\mathcal{H}_{as} & \mathcal{H}_{as}^2 + \dfrac{\overline{\mathcal{W}_{as}^2}}{\overline{\Theta_{ss}^2}} & \mathcal{H}_{as}\mathcal{H}_{sa} & -\mathcal{H}_{as}\mathcal{H}_{ss} \\
-\mathcal{H}_{aa}\mathcal{H}_{sa} & \mathcal{H}_{as}\mathcal{H}_{sa} & \mathcal{H}_{sa}^2 + \dfrac{\overline{\mathcal{W}_{sa}^2}}{\overline{\Theta_{ss}^2}} & -\mathcal{H}_{sa}\mathcal{H}_{ss} \\
\mathcal{H}_{aa}\mathcal{H}_{ss} & -\mathcal{H}_{as}\mathcal{H}_{ss} & -\mathcal{H}_{sa}\mathcal{H}_{ss} & \mathcal{H}_{ss}^2 + \dfrac{\overline{\mathcal{W}_{ss}^2}}{\overline{\Theta_{ss}^2}} \\
\mathcal{H}_{as} & \mathcal{H}_{aa} & -\mathcal{H}_{ss} & -\mathcal{H}_{sa} \\
\mathcal{H}_{sa} & -\mathcal{H}_{ss} & \mathcal{H}_{aa} & -\mathcal{H}_{as} \\
\mathcal{H}_{ss} & \mathcal{H}_{sa} & \mathcal{H}_{as} & \mathcal{H}_{aa}
\end{bmatrix}
\begin{bmatrix}
\mathcal{F}_{aa} \\ \mathcal{F}_{as} \\ \mathcal{F}_{sa} \\ \mathcal{F}_{ss}
\end{bmatrix}
=
\begin{bmatrix}
\mathcal{H}_{aa} \\ -\mathcal{H}_{as} \\ -\mathcal{H}_{sa} \\ \mathcal{H}_{ss} \\ 0 \\ 0 \\ 0
\end{bmatrix}
\tag{132}
$$

## *Bibliography*


1. Ahmed, N., T. Natarajan, and K.R. Rao. "Discrete Cosine Transform," *IEEE Transactions on Computers, C-23*: 90-93 (January 1974).

2. Ayers, G.R. and J.C. Dainty. "Iterative Blind Deconvolution Method and its Applications," *Optics Letters, 13*: 547-549 (July 1988).

3. Bongiovanni, G., P. Corsini, and G. Frosini. "One Dimensional and Two-Dimensional Generalized Discrete Fourier Transforms," *IEEE Transactions on Acoustics, Speech, and Signal Processing, ASSP-24*: 97-99 (February 1976).

4. Cannon, M. "Blind Deconvolution of Spatially Invariant Image Blurs with Phase," *IEEE Transactions on Acoustics, Speech, and Signal Processing, ASSP-24*: 58-63 (February 1976).

5. Chen, W.H., C.H. Smith, and S.C. Fralick. "A Fast Computational Algorithm for the Discrete Cosine Transform," *IEEE Transactions on Communications, COMM-25*: 1004-1009 (September 1977).

6. Chitprasert, B. and K.R. Rao. "Discrete Cosine Transform Filtering," *Signal Processing, 19*: 233-245 (March 1990).

7. Davenport, W.B. *Probability and Random Processes: An Introduction for Applied Scientists and Engineers.* New York: McGraw-Hill, 1970.

8. Davey, B.K., R.G. Lane, and R.T. Bates. "Blind Deconvolution of Noisy Complex-Valued Image," *Optics Communications, 69*: 353-356 (January 15, 1989).

9. Foltz, T.M. and B.M. Welsh. "Symmetric Convolution of Asymmetric Multidimensional Sequences Using Discrete Trigonometric Transforms," submitted to *IEEE Transactions on Image Processing*, October 1997; accepted with minor revisions, May 1998.

10. _____. "Image Reconstruction Using Symmetric Convolution and Discrete Trigonometric Transforms," submitted to the *Journal of the Optical Society of America A*, March 1998.

11. _____. "Diagonal Forms of Symmetric Convolution Matrices for Asymmetric Multidimensional Sequences," to be presented at *Applications of Digital Image Processing XXI*, during SPIE's 1998 Annual Meeting (San Diego, CA), July 19-24, 1998.

12. _____. "A Scalar Wiener Filter Based on Discrete Trigonometric Transforms and Symmetric Convolution," to be presented at *Propagation and Imaging Through the Atmosphere II*, during SPIE's 1998 Annual Meeting (San Diego, CA), July 19-24, 1998.

13. Ford, S.D. *Linear Reconstruction of Non-Stationary Image Ensembles Incorporating Blur and Noise Models*. Ph.D. Dissertation, AFIT/DS/ENG/98-02. Graduate School of Engineering, Air Force Institute of Technology (AETC), Wright-Patterson AFB OH, March 1998.

14. _____, B.M. Welsh, M.C. Roggemann, and D.J. Lee. "Reconstruction of Low-Light Images by use of the Vector Wiener Filter," *Journal of the Optical Society of America A, 14*: 2678-2691 (October 1997).

15. Gonzalez, R.C. and R.E. Woods. *Digital Image Processing*. Reading, MA: Addison-Wesley, 1992.

16. Goodman, J.W. *Fourier Optics*. New York: McGraw-Hill, 1968.

17. _____. *Statistical Optics*. New York: John Wiley and Sons, 1985.

18. Graham, A. *Kronecker Products and Matrix Calculus with Applications*. New York: John Wiley and Sons, 1981.

19. Hahn, S.L. "Multidimensional Complex Signals with Single-Orthant Spectra," *Proceedings of the IEEE, 80*: 1287-1300 (August 1992).

20. Hamidi, M. and J. Pearl. "Comparison of the Cosine and Fourier Transforms of Markov-1 Signals," *IEEE Transactions on Acoustics, Speech, and Signal Processing, ASSP-24*: 428-429 (October 1976).

21. Helstrom, C.W. "Image Restoration by the Method of Least Squares," *Journal of the Optical Society of America, 57*: 297-303 (March 1967).

22. Horn, R.A. and C.R. Johnson. *Topics in Matrix Analysis*. New York: Cambridge University Press, 1991.

23. Hunt, B.R. "A Matrix Theory Proof of the Discrete Convolution Theorem," *IEEE Transactions on Audio and Electroacoustics, AU-19*: 285-288 (December 1971).

24. _____. "The Application of Constrained Least Squares Estimation to Image Restoration by Digital Computer," *IEEE Transactions on Computers, C-22*: 805-812 (September 1973).

25. _____ and T.M. Cannon. "Nonstationary Assumptions for Gaussian Models of Images," *IEEE Transactions on Systems, Man, and Cybernetics, SMC-6*: 876-882 (December 1976).

26. Jain, A.K. "A Sinusoidal Family of Unitary Transforms," *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-1*: 356-365 (October 1979).

27. _____. *Fundamentals of Digital Image Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1989.

28. Kay, S.M. *Fundamentals of Statistical Signal Processing.* Englewood Cliffs, NJ: Prentice-Hall, 1993.

29. Kekre, H.B. and J.K. Solanki. "Comparative Performance of Various Trigonometric Unitary Transforms for Image Coding," *International Journal of Electronics, 44*: 305-315 (March 1978).

30. Knox, K.T. and B.J. Thompson. "Recovery of Images from Atmospherically Degraded Short Exposure Images," *Astrophysics Journal, 193*: L45-L48 (October 1, 1974).

31. Labeyrie, A. "Attainment of Diffraction Limited Resolution in Large Telescopes by Fourier Analyzing Speckle Patterns in Star Images," *Astronomy and Astrophysics, 6*: 85-87 (May 1970).

32. Lohman, A., W.G. Weigelt, and B. Wirnitzer. "Speckle Masking in Astronomy: Triple Correlation Theory and Applications," *Applied Optics, 22*: 4028-4037 (December 15, 1983).

33. Malvar, H.S. *Signal Processing with Lapped Transforms.* Norwood, MA: Artech, 1992.

34. Martucci, S.A. "Symmetric Convolution and the Discrete Sine and Cosine Transforms," *IEEE Transactions on Signal Processing, SP-42*: 1038-1051 (May 1994).

35. _____. "Digital Filtering of Images Using the Discrete Sine and Cosine Transforms," *Optical Engineering, 35*: 119-127 (January 1996).

36. Matson, C.L. "Fourier Spectrum Extrapolation and Enhancement Using Support Constraints," *IEEE Transactions on Signal Processing, SP-42*: 156-163 (January 1994).

37. Oppenheim, A.V. and R.W. Schafer. *Digital Signal Processing.* Englewood Cliffs, NJ: Prentice-Hall, 1975.

38. Pratt, W.K. "Generalized Wiener Filter Computation Techniques," *IEEE Transactions on Computers, C-21*: 636-641 (July 1972).

39. Rao, K.R. and P. Yip. *Discrete Cosine Transform: Algorithms, Advantages, Applications.* San Diego: Academic Press, 1990.

40. Roggemann, M.C. and B.M. Welsh. *Imaging Through Turbulence.* Boca Raton, FL: CRC Press, 1996.

41. Sánchez, V., P. García, A.M. Peinado, J.C. Segura, and A.J. Rubio. "Diagonalizing Properties of the Discrete Cosine Transforms," *IEEE Transactions on Signal Processing, SP-43*: 2631-2641 (November 1995).

42. Sánchez, V., A.M. Peinado, J.C. Segura, P. García, and A.J. Rubio. "Generating Matrices for the Discrete Sine Transforms," *IEEE Transactions on Signal Processing, SP-44*: 2644-2646 (October 1996).

43. Scharf, L.L. *Statistical Signal Processing: Detection, Estimation, and Time Series Analysis*. Reading, MA: Addison-Wesley, 1991.

44. Slepian, D. "Linear Least-Squares Filtering of Distorted Images," *Journal of the Optical Society of America, 57*: 918-922 (July 1967).

45. Stockham, J.T.G., T.M. Cannon, and R.B. Ingebretsen. "Blind Deconvolution Through Digital Signal Processing," *Proceedings of the IEEE, 63*: 678-692 (April 1975).

46. Strang, G. *Linear Algebra and its Applications* (Third edition). Orlando, FL: Harcourt Brace Jovanovich, 1988.

47. Stuller, J.A. and B. Kurz. "Two-Dimensional Markov Representations of Sampled Images," *IEEE Transactions on Communications, COMM-24*: 1148-1152 (October 1976).

48. Therrien, C.W. *Discrete Random Signals and Statistical Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1992.

49. Vernet, J.L. "Real Signals Fast Fourier Transform: Storage Capacity and Step Number Reduction by Means of an Odd Discrete Fourier Transform," *Proceeding of the IEEE, 59*: 1531-1532 (October 1971).

50. Wang, Z. "Fast Algorithms for the Discrete W Transform and for the Discrete Fourier Transform," *IEEE Transactions on Acoustics, Speech, and Signal Processing, ASSP-32*: 803-816 (August 1984).

51. Whiteley, M.R., B.M. Welsh, and M.C. Roggemann. "Limitations of Gaussian Assumptions for the Irradiance Distribution in Digital Imagery: Nonstationary Image Ensemble Considerations," *Journal of the Optical Society of America A, 15*: 802 - 810 (April 1998).

52. Wiener, N. *Extrapolation, Interpolation, and Smoothing of Stationary Time Series*. New York: John Wiley and Sons, 1949.

53. Woods, J.W. "Two-Dimensional Discrete Markovian Fields," *IEEE Transactions on Information Theory, IT-18*: 232-240 (March 1972).

54. Zwillinger, D., ed.-in-chief. *CRC Standard Mathematical Tables and Formulae* (Thirtieth Edition). Boca Raton, FL: CRC Press, 1996.

## *Vita*

Major Thomas Mark Foltz was born on 12 May 1962 in Hershey, Pennsylvania. He graduated form Wilde Lake High School in Columbia, Maryland in 1980 and received a Bachelor of Electrical Engineering degree from the Georgia Institute of Technology in 1984. After receiving a commission through AFROTC, he attended Communications-Electronics training at Keesler AFB, Mississippi. He was then assigned to the 485th Engineering Installation Group, Griffiss AFB, New York. He graduated from AFIT in 1988 with a Master of Science in Electrical Engineering. His next assignments were to the 84th Radar Evaluation Squadron, Hill AFB, Utah, and the Air Force Military Personnel Center, Randolph AFB, Texas. He entered the AFIT Ph.D. program in June 1995 and will continue his studies at Air Command and Staff College, Maxwell AFB, Alabama, after graduation from AFIT.

Permanent Address: 11901 Yellow Rush Pass
Columbia, MD 21044-4508

# REPORT DOCUMENTATION PAGE

*Form Approved*
*OMB No. 0704-0188*

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

| 1. AGENCY USE ONLY *(Leave blank)* | 2. REPORT DATE | 3. REPORT TYPE AND DATES COVERED |
|---|---|---|
| | 1 Jun 98 | Dissertation |

**4. TITLE AND SUBTITLE**
Trigonometric Transforms for Image Reconstruction

**5. FUNDING NUMBERS**

**6. AUTHOR(S)**
Thomas M. Foltz

**7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)**
Air Force Institute of Technology
AFIT/ENG
2950 P St., Bldg. 640
Wright-Patterson AFB, OH 45433-7765

**8. PERFORMING ORGANIZATION REPORT NUMBER**
AFIT/DS/ENG/98-04

**9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)**
Capt Bruce Stribling
Air Force Maui Optical Station (AMOS), Phillips Laboratory, PL/LIMA
535 Lipoa Pkwy, Ste. 200
Kihei, HI 96753 (808) 875-8390

**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**

**11. SUPPLEMENTARY NOTES**

**12a. DISTRIBUTION AVAILABILITY STATEMENT**
Approved for public release; distribution unlimited.

**12b. DISTRIBUTION CODE**

**13. ABSTRACT** *(Maximum 200 words)*

This dissertation demonstrates how the symmetric convolution-multiplication property of discrete trigonometric transforms can be applied to traditional problems in image reconstruction with slightly better performance than Fourier techniques and increased savings in computational complexity for symmetric point spread functions. The fact that the discrete Fourier transform diagonalizes a circulant matrix provides an alternate way to derive the symmetric convolution-multiplication property for discrete trigonometric transforms. Derived in this manner, the sym-metric convolution-multiplication property extends easily to multiple dimensions and generalizes to multidimensional asymmetric sequences. The symmetric convolution-multiplication property allows for linear filtering of degraded images via point-by-point multiplication in the transform domain of trigonometric transforms. Specifically in the transform domain of a type-II discrete cosine transform, there is an asymptotically optimum energy compaction about the low-frequency indices of highly correlated images which has advantages in reconstructing images with high-frequency noise. The symmetric convolution-multiplication property allows for well-approximated scalar representations in the trigonometric transform domain for linear reconstruc-tion filters such as the Wiener filter. An analysis of the scalar Wiener filter's improved mean-squared error performance in the trigonometric transform domain is given.

**14. SUBJECT TERMS**
Trigonometric transforms, multidimensional filters, inverse filters, Wiener filters, image reconstruction

**15. NUMBER OF PAGES**
148

**16. PRICE CODE**

| 17. SECURITY CLASSIFICATION OF REPORT | 18. SECURITY CLASSIFICATION OF THIS PAGE | 19. SECURITY CLASSIFICATION OF ABSTRACT | 20. LIMITATION OF ABSTRACT |
|---|---|---|---|
| UNCLASSIFIED | UNCLASSIFIED | UNCLASSIFIED | UL |